

*Citation for published version:*

Zhen, L, Hu, Y, Wang, S, Laporte, G & Wu, Y 2019, 'Fleet deployment and demand fulfillment for container shipping liners', *Transportation Research Part B: Methodological*, vol. 120, pp. 15-32.  
<https://doi.org/10.1016/j.trb.2018.11.011>

*DOI:*

[10.1016/j.trb.2018.11.011](https://doi.org/10.1016/j.trb.2018.11.011)

*Publication date:*

2019

*Document Version*

Peer reviewed version

[Link to publication](#)

*Publisher Rights*

CC BY-NC-ND

**University of Bath**

**Alternative formats**

If you require this document in an alternative format, please contact:  
[openaccess@bath.ac.uk](mailto:openaccess@bath.ac.uk)

**General rights**

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

**Take down policy**

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

# Fleet deployment and demand fulfillment for container shipping liners

Lu Zhen<sup>a</sup>, Yi Hu<sup>a</sup>, Shuaian Wang<sup>b,\*</sup>, Gilbert Laporte<sup>c</sup>, Yiwei Wu<sup>a</sup>

<sup>a</sup>*School of Management, Shanghai University, Shanghai, China*

<sup>b</sup>*The Hong Kong Polytechnic University Shenzhen Research Institute, Nanshan District, Shenzhen, China*

*Department of Logistics & Maritime Studies, The Hong Kong Polytechnic University, Hung Hom, Hong Kong*

<sup>c</sup>*Department of Decision Sciences, HEC Montréal, Montréal, Canada*

---

## Abstract

This paper models and solves a fleet deployment and demand fulfillment problem for container shipping liners with consideration of the potential overload risk of containers. Given the stochastic weights of transported containers, chance constraints are embedded in the model at the strategic level. Several realistic limiting factors such as the fleet size and the available berth and yard resources at the ports are also considered. A non-linear mixed integer programming (MIP) model is suggested to optimally determine the transportation demand fulfillment scale for each origin-destination pair, as well as the ship deployment plan along each route, with an objective incorporating revenue, fixed operation cost, fuel consumption cost, holding cost for transshipped containers, and extra berth and yard costs. Two efficient algorithms are then developed to solve the non-linear MIP model for different instance sizes. Numerical experiments based on real-world data are conducted to validate the effectiveness of the model and the algorithms. The results indicate the proposed methodology yields solutions with an optimality gap less than about 0.5%, and can solve realistic instances with 19 ports and four routes within about one hour.

**Keywords:** Demand fulfillment; fleet deployment; transshipment; port capacity; stochastic container weight.

---

\*Corresponding author. lzhen@shu.edu.cn (L. Zhen), wangshuaian@gmail.com (S. Wang)

## 1. Introduction

Shipping liners play an important role in today's economy which is becoming increasingly global, and more operations are being outsourced and moved offshore (Fransoo and Lee, 2013). Shipping liners run weekly-serviced ship routes with fixed schedules to transport containers for customers. Each shipping company operates its own shipping network covering a number of routes (services) and ports. A shipping liner cannot usually fulfill all customer demands in a given container transportation market due to the limitations of its fleet size and of the available port resources (e.g., berths and yard space), and because of some other unforeseen factors (Zhen, 2015, 2016). The transportation demand is usually characterized by the number of containers that need to be transported between the origin-destination (OD) pairs of the shipping network. Given the data on the full-size market demand, a shipping liner needs to determine an economic fulfillment scale for each OD pair's transportation demand, as well as the number of ships deployed on each route of its shipping network so as to maximize its profit. This is an important strategic decision for the managers of shipping liners.

The above strategic level problem involves intertwined decisions as well as numerous complex factors. While it is easy to understand that the demand fulfillment scale is positively related to the number of deployed ships, the optimal allocation of the available ships along the routes is not a straightforward decision because of the different unit transportation fees among OD pairs, the different cost configurations among the routes, and the complex underlying relationship between the OD pairs and the routes. A liner may not always fulfill as much transportation demand as possible by using all its available ships because the port resources reserved for the liner in the shipping network are fixed. Moreover, several features proper to the ocean shipping industry must also be considered in this strategic level decision problem. For example, the number of ships deployed on a route affects the ships' speed on each leg of a route, which further influences fuel consumption and cost. These costs and the fixed operation costs of the deployed ships jointly constitute the bulk of the cost for a shipping liner. In addition, the ship schedule of each route (service) affects the containers' storage time at the transshipment hubs which connect the routes in the shipping network. The holding cost of the transshipped containers should therefore

also be taken into account. Finally, the potential overload risk of containers should not be ignored since the weights of the transported containers are stochastic. Wang et al. (2016) state that almost all the existing literature regards the weights of containers as constants and few existing studies consider the problem of container overload. However, the potential overload risk of containers occurs frequently and has irreparable consequences. Indeed, ship overload accidents account for 60 percent of accidents on inland waterways and up to 70 percent in some areas. Therefore, studying the overload risk of containers is practical. For example, a shipping liner may promise a quota of 1,000 twenty-foot equivalent units (TEUs) to a customer with respect to an OD pair, but when the shipping liners make long term decisions on the demand fulfillment scale, the cargo types and weights in the containers are unforeseen. For example, the weights of 1,000 TEUs of plastic and of metal are significantly different. The overload risk should therefore also be controlled.

This paper provides a comprehensive study of this complex decision problem. Given a shipping network with multiple routes connected by transshipment hubs and the transportation demand information, we propose a non-linear chance-constrained mathematical integer program (MIP) to optimally determine the transportation demand fulfillment scale for each OD pair, as well as the ship deployment plan along each route in order to maximize the total profit, equal the revenue earned by fulfilling the demand, minus four types of cost: the fixed operation cost of the deployed ships, the fuel cost, the cost for storing transshipped containers at ports, and the cost of using extra port resources. The chance constraints embedded in the model control the potential overload risk resulting from random container weights. In addition to the chance constraints, the model contains other non-linear components. Some new techniques are suggested to linearize the model into a mixed integer second-order cone programming (MISOCP) model that can be tractable for some commercial solvers such as CPLEX.

The remainder of this paper is organized as follows. Section 2 provides an overview of the related literature. Section 3 describes the problem. A mathematical model is proposed in Section 4, followed by a linearization scheme in Section 5. Two heuristics are developed to solve the model in Section 6. Section 7 reports the results of our computational experiments. Closing remarks and conclusions follow in Section 8.

## 2. Related works

There exist numerous related studies on fleet deployment. Readers interested in overviews can refer to Ronen (1993), Christiansen et al. (2004), Christiansen et al. (2013), and Meng et al. (2014). At the strategic decision level, the fleet deployment problem (FDP) consists of assigning available vessels to predetermined voyages (Fagerholt et al., 2009) in order to maximize profit or minimize cost.

Several linear programming and mixed integer linear programming models for the FDP have been put forward. Perakis and Jaramillo (1991) were the first to develop a linear programming model for the FDP, which takes account of ship capacity, and minimizes service frequency requirements as well as ship charter cost. However, this model works with continuous decision variables for the allocation of ships to shipping routes, instead of integer variables. To remedy this problem, Jaramillo and Perakis (1991) proposed an integer programming model. Cho and Perakis (1996) formulated a MIP model for the FDP, where the demand of containers between two given ports can be served by any shipping route passing through the two ports. Powell and Perkins (1997) extended the model of Jaramillo and Perakis (1991) by adding ship lay-out costs to the objective function. Álvarez (2009) proposed a MIP formulation for the integrated optimization of vessel routing and fleet deployment. Based on previous works, Gelareh and Meng (2010) developed a MIP model for the FDP in which speed is a decision variable, and investigated the problem of ship speed optimization to obtain optimal sailing speeds through a non-linear model, which can be approximated as a MIP model. This model was later improved by Wang et al. (2011). Meng and Wang (2011) investigated a multi-period fleet planning and FDP with a known container demand for each OD pair and each period. Meng and Wang (2010) proposed a chance-constrained model for the FDP under uncertain demand, but ignored transshipment activities. Because the speed of ships has an impact on fuel consumption cost, Zacharioudakis et al. (2011) developed a practical methodology that considers the effect of speed on fuel consumption for shipping companies to solve FDPs. Andersson et al. (2015) put forward an integrated model to optimize fleet deployment and sailing speed for RoRo shipping companies. Zheng et al. (2015) set up a shipping network for liner shipping alliances, and proposed a model with consideration of ship deployment, cargo allocation, and container routing. Xia et al.

109 (2015) developed a comprehensive model to simultaneously and optimally determine  
110 ship deployment, sailing speed, and container allocation in order to maximize profit  
111 at the strategic level. Zhao et al. (2016) designed a novel method of fleet deployment  
112 based on risk evaluation so as to take advantage of resources for navigation and reduce  
113 risks. Monemi and Gelareh (2017) provided an integrated model considering shipping  
114 network design, FDP and empty container repositioning. The number of routes and  
115 their design play an endogenous role in their problem. Wang et al. (2017) proposed  
116 a two-stage stochastic programming model to optimally solve the FDP and compute  
117 the sailing speeds with the consideration of market uncertainties. Some studies have  
118 incorporated container transshipment in FDPs. Wang and Meng (2012) developed  
119 an MIP model for the FDP in which containers can be transshipped at any port,  
120 which was extended by Meng and Wang (2012) by adding transit time constraints.

121 There also exist some studies on FDPs that consider the uncertainties of liner ser-  
122 vice schedule or container shipment demand. Wang and Meng (2012), Qi and Song  
123 (2012) and Bell et al. (2011) considered uncertainty in the liner service schedule but  
124 ignored uncertainty in container shipment demand. In order to tackle demand uncer-  
125 tainty, Meng and Wang (2010) proposed a chance-constrained model, which extends  
126 the deterministic FDP to a FDP under uncertainty. Meng et al. (2012) assumed that  
127 the container shipment demand is a random variable, and hence formulated a two-  
128 stage stochastic integer programming model, and developed an algorithm integrating  
129 sample average approximation with a dual decomposition and Lagrangian relaxation  
130 method. Wang et al. (2012) further extended the model of Meng et al. (2012) by  
131 adding the expectation and variance of the cost in the objective function.

132 In conclusion, several related studies on the FDP have not taken transshipment  
133 activities into account. Although some authors did consider these, they did not incor-  
134 porate the demand fulfillment decision and the potential overload risk of containers  
135 due to their stochastic weights. Moreover, some port resources such as berths and  
136 yard space, which are crucial in maritime activities, have also been ignored. (Liu  
137 et al., 2016) conducted an integrated planning of the berth allocation and the yard  
138 allocation in container terminals.

139 Our paper proposes an integrated decision model that compounds ship fleet de-  
140 ployment and demand fulfillment decisions by considering crucial factors such as  
141 transshipment activities, the stochastic weight of containers, port resources, the

timetabling of ship visits at each port of call, and the demand fulfillment scale for each OD pair. There is no doubt that these factors complicate this already difficult fleet deployment and demand fulfillment problem. We propose a comprehensive model and we develop some techniques to handle the complexity resulting from the chance constraints. We believe the problem features considered in our study are realistic and new with respect to previous research.

### 3. Problem description

We consider a shipping liner operating on a network containing a set  $R$  of container shipping routes (services), which cover a set  $P$  of ports. Figure 1 depicts a shipping network with four routes and 19 ports. Each ship route  $r$  is described as (port  $p_{r1}$ , port  $p_{r2}, \dots$ , port  $p_{ri}, \dots$ , port  $p_{rN_r}$ , port  $p_{r1}$ ), which implies that ship route  $r$  has  $N_r$  ports of call as well as  $N_r$  legs. Let leg  $i$  denote the voyage from port  $p_{ri}$  to port  $p_{r,i+1}$ , where  $p_{r,N_r+1} = p_{r1}$ . We denote by  $I_r$  the set of legs in ship route  $r$ . The details on the objective and key constraints considered in this study are provided in the following subsections.

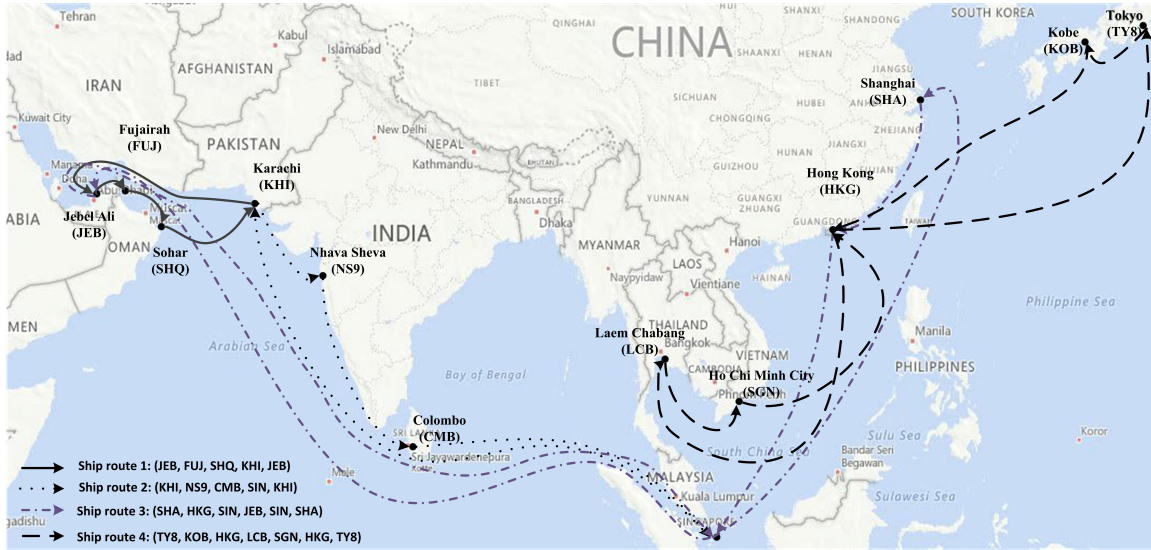


Figure 1: A shipping network with four routes

### 157 3.1. Revenue of the demand fulfillment

158 Container transportation demand is usually described by OD pairs indexed by  
 159  $\varepsilon \in \Omega$ . The number of containers requesting transport for each OD pair during a  
 160 week can be estimated according to historical data. Given the unit fee for transporting  
 161 a TEU container, we can compute the maximum revenue  $V_\varepsilon$  that can be earned if all  
 162 the transportation demand of OD pair  $\varepsilon$  is fulfilled. We define a variable  $\pi_\varepsilon$  equal to  
 163 the percentage of OD pair  $\varepsilon$ 's transportation demand fulfilled by the shipping liner.  
 164 Then the total revenue can be calculated as  $\sum_{\varepsilon \in \Omega} V_\varepsilon \pi_\varepsilon$ .

### 165 3.2. Fixed operation cost of deployed ships

166 A fleet of homogeneous ships is deployed on each route to maintain a weekly  
 167 service frequency. If the number of ships deployed on route  $r$  is  $\beta_r$ , then the total  
 168 fixed operation cost for all the deployed ships in all the routes during one week can  
 169 be calculated as  $\sum_{r \in R} C_r^{Opr} \beta_r$ , where  $C_r^{Opr}$  is the weekly operation cost for deploying  
 170 one ship on route  $r$ .

### 171 3.3. Fuel cost depending on sailing speed

172 The total time for a ship completing the travel along a route is  $7\beta_r$  days. More  
 173 specifically,  $\sum_{i \in I_r} (d_{ri} + \delta_{ri}) = 7\beta_r$ , where  $d_{ri}$  is the dwell time of ships at the  $i^{th}$   
 174 port of call on ship route  $r$ , and  $\delta_{ri}$  is the sailing time of ships on the  $i^{th}$  leg on ship  
 175 route  $r$ . In reality, the port dwell time  $d_{ri}$  is usually predetermined according to some  
 176 contracts between the shipping liner and port operators, but the sailing time  $\delta_{ri}$  of  
 177 each leg can be a decision variable for the shipping liner, which can be used to modify  
 178 the value of  $\delta_{ri}$  by updating the ships' speed on each leg.

179 A ship's unit fuel consumption significantly depends on its sailing speed. In this  
 180 study, we assume that the unit fuel consumption function on sailing speed  $y$  is cal-  
 181 culated as  $y = kx^a$  (USD per nautical mile), where  $x$  is the speed, and  $k$  and  $a$  are  
 182 positive coefficients. More specifically, the fuel cost for the  $i^{th}$  leg on ship route  $r$  is  
 183  $l_{ri} k_{ri} (l_{ri} / \delta_{ri})^{a_{ri}}$ , where  $l_{ri}$  is the leg's length, and  $k_{ri}$  and  $a_{ri}$  are coefficients that can  
 184 be estimated according to historical data. The total fuel cost is then calculated as  
 185  $\sum_{r \in R} \sum_{i \in I_r} l_{ri} k_{ri} (l_{ri} / \delta_{ri})^{a_{ri}}$ .



### 186 3.4. Holding cost for storing transshipped containers

187 The above decisions on ship deployment and sailing speeds influence the cost  
 188 related to each route which are inter-route costs. Decisions made on the arrival time  
 189 of ships at each port of call in each route affect the storing time and cost of the  
 190 containers at transshipment hubs, which are inter-route costs.

191 We define a quadruple  $(r, i, s, j)$  to denote that the  $i^{th}$  port of call on ship route  
 192  $r$  and the  $j^{th}$  port of call on ship route  $s$  are the same physical port in the network,  
 193 where  $r, s \in R, i \in I_r$  and  $j \in I_s$ . Hence  $Q = \{(r, i, s, j) | p_{ri} = p_{sj}\}$ . Let  $m_{risj\varepsilon}$  be  
 194 the maximum number of TEUs transshipped at hub  $(r, i, s, j)$  for OD pair  $\varepsilon$  if all  
 195 the transportation demand for the OD pair is fulfilled. Then the number of trans-  
 196 shipped containers at the hub is  $\pi_\varepsilon m_{risj\varepsilon}$ . We define a parameter  $C^{Hold}$  equal to the  
 197 unit holding cost (USD per TEU per day), and a variable  $\gamma_{risj}$  to denote the dif-  
 198 ference in days between the time a ship visits the port of call  $(r, i)$  and the time a  
 199 ship visits  $(s, j)$ . Then the total holding cost for storing transshipped containers is  
 200  $C^{Hold} \sum_{(r,i,s,j) \in Q} \sum_{\varepsilon \in \Omega} \pi_\varepsilon m_{risj\varepsilon} \gamma_{risj}$ .

### 201 3.5. Cost for using extra berth or yard space

202 Each port has a certain yard space reserved for storing transshipped containers,  
 203 and a certain number of berths for the shipping liner, booked in advance according  
 204 to contracts. If the yard space and berth capacity limitations at ports are violated,  
 205 then some extra costs are incurred (Petering et al., 2017).

206 In this study, we define  $B_p$  as the set of berths  $b$  in port  $p$  reserved for the shipping  
 207 liner. Another index  $\hat{b}$  is defined as a dummy berth, which is used when there are no  
 208 available berths in the reserved berth set  $B_p$  when a ship arrives at port  $p$ . From the  
 209 perspective of modeling, if the dummy berth  $\hat{b}$  is used by a ship, then an extra cost  
 210 is incurred. Here we define binary decision variables  $\theta_{rib}$  to denote whether the ship  
 211 arrives at berth  $b$  in the port of call  $(r, i)$ , and we define a parameter  $C_{pri}^{Berth}$  as the  
 212 penalty cost incurred when the dummy berth  $\hat{b}$  is used in the port of call  $(r, i)$ . Then  
 213 the total cost for extra berth usage is  $\sum_{r \in R} \sum_{i \in I_r} C_{pri}^{Berth} \theta_{ri\hat{b}}$ .

214 For the yard resource, we also define an auxiliary variable  $\lambda_{pw}$  as the extra used  
 215 yard space (measured in number of TEUs) for storing transshipped containers at  
 216 port  $p$  on day  $w \in W$  of a week. The formula for computing the variable  $\lambda_{pw}$  will be

explained in the Section 4. Let  $C_p^{Yard}$  be the penalty cost for using one unit of extra yard space (TEUs), beyond the agreed reserved yard space, in port  $p$  to store the transshipped containers for one day. Then the total cost for extra yard space usage is  $\sum_{p \in P} \sum_{w \in W} C_p^{Yard} \lambda_{pw}$ .

### 3.6. Risk of overload due to random container weight

This study also considers the potential overload risk of ships due to the stochastic weight of transported containers. To illustrate this, suppose a liner promises a customer or an agency a quota of one thousand TEUs for an OD pair  $\varepsilon$ . When the liner makes the long term decision on the demand fulfillment scale for that OD pair, the weights of the cargos in the containers are unforeseen and may create an overload. We define a parameter  $n_{ri\varepsilon}$  as the maximum number of containers transported on leg  $(r, i)$  for OD pair  $\varepsilon$  if all the transportation demand for the OD pair is fulfilled. Thus there will be  $\lceil \pi_\varepsilon n_{ri\varepsilon} \rceil$  containers be transported on leg  $(r, i)$  for the OD pair  $\varepsilon$ . A stochastic parameter  $\tilde{c}_{ri\varepsilon u}$  is defined as the random weight of the containers on leg  $(r, i)$  for OD pair  $\varepsilon$ , where  $u$  is the index of the container. Suppose the maximum load capacity (in tons) of a ship on leg  $(r, i)$  is  $A_{ri}^{Load}$ , and the probability of overload should be constrained to lie under a level  $\alpha$  (e.g., 1%, 0.1%), then the constraint  $\text{Prob}(\sum_{\varepsilon \in \Omega} \sum_{u=1}^{\lceil \pi_\varepsilon n_{ri\varepsilon} \rceil} \tilde{c}_{ri\varepsilon u} > A_{ri}^{Load}) \leq \alpha$  should hold for each leg  $(r, i)$ .

### 3.7. Assumptions and data preparation before using the model

Based on the above analysis on the revenue, on the various types of costs considered in the objective function and on the chance constraints controlling the overload risk, we will formulate a mathematical model in the next section. We first clarify the assumptions of this study:

- (1) the shipping network of the ports and routes (voyages) is already determined;
- (2) the ships are homogenous on each route in terms of capacity and cost structure;
- (3) the ships' dwell time at each port of call is deterministic.

Finally, we provide some explanation on how to prepare some key input data for the decision model. First, a shipping liner should collect the historical data on the weekly demand for each OD pair (Fagerholt et al., 2009; Bell et al., 2013). Based on the estimated unit price for shipping one TEU for each OD pair  $\varepsilon$ , the liner can calculate the  $V_\varepsilon$  values (i.e., the maximum revenue that can be earned if all the

248 transportation demand of the OD pair  $\varepsilon$  is fulfilled). Moreover, the mapping from  
 249 an OD pair to a set of its covered legs as well as a set of transshipment ports is  
 250 also deterministic. Given this mapping information, the liner can also estimate the  
 251 parameters  $n_{ri\varepsilon}$  (i.e., the maximum number of containers transported on each leg  
 252  $(r, i)$  for each OD pair  $\varepsilon$ ) and the parameters  $m_{risj\varepsilon}$  (i.e., the maximum number of  
 253 containers transshipped from the port of call  $(r, i)$  to  $(s, j)$  for OD pair  $\varepsilon$ , if all the  
 254 demand is fulfilled).

255 Another important input value is the stochastic parameter  $\tilde{c}_{ri\varepsilon u}$  about the ran-  
 256 dom container weight on the leg  $(r, i)$  for the OD pair  $\varepsilon$ . The liner could collect the  
 257 historical data on the weights of containers transported for each OD pair  $\varepsilon$ , and then  
 258 calibrate the expected value and standard deviation. Given the mapping information  
 259 between an OD pair and the set of its covered legs, one can obtain the expected value  
 260  $\mu_{ri\varepsilon}$  and the standard deviation  $\sigma_{ri\varepsilon}$  for the random weights of containers transported  
 261 on each leg  $(r, i)$  for each OD pair  $\varepsilon$ . These two parameters will be used in Section 5  
 262 to linearize the chance constraints in the model.

## 263 4. Model formulation

264 We now introduce a non-linear chance-constrained MIP model for the problem.  
 265 We first define some indices, sets, input parameters and decision variables.

### 266 Indices and sets

267	$\varepsilon$	index of an OD pair;
268	$\Omega$	set of all the OD pairs;
269	$r$ (or $s$ )	index of a ship route;
270	$R$	set of all the ship routes;
271	$i$ (or $j$ )	index of port of call (or leg) on a ship route (leg $i$ is from port of call $i$ to $i+1$ );
272	$I_r$	set of the ports of call (or legs) on ship route $r$ ;
273	$p$	index of a physical port, which is different from the port of call (defined as $i$ );

274	$P$	set of all the ports;
275	$p_{ri}$	index of the port, which corresponds to the port of call $(r, i)$ ;
276	$I'_{rp}$	set of the ports of call (or legs) on ship route $r$ ; these port of calls are the same physical port $p$ ;
277	$R'_p$	set of ship routes that include port $p$ ;
278	$Q$	set of quadruples $(r, i, s, j)$ , where $r, s \in R; i \in I_r, j \in I_s$ ; a $(r, i, s, j)$ means the ports of call $(r, i)$ and $(s, j)$ are the same physical port in shipping network. $Q = \{(r, i, s, j)   p_{ri} = p_{sj}\}$ ;
279	$Q_p$	a subset of $Q$ ; $Q_p = \{(r, i, s, j)   p_{ri} = p_{sj} = p\}$ ;
280	$w$	index of a day in a week, i.e., $0 = \text{Sun}, 1 = \text{Mon}, 2 = \text{Tue}, \dots, 6 = \text{Sat}$ ;
281	$W$	set of days in a week, $W = \{0, 1, 2, \dots, 6\}$ ;
282	$b$	index of a berth;
283	$B_p$	set of berths in port $p$ ; these berths are reserved for the shipping liner;
284	$\hat{b}$	index of a dummy berth, used when there are not available berths in the reserved berth set $B_p$ when a ship arrives at port $p$ ;
285	$\mathbb{Z}$	set of integers;
286	$\mathbb{Z}_+$	set of non-negative integers.

287 **Parameters**

	$V_\varepsilon$	maximum revenue if all the transportation demand of OD pair $\varepsilon$ is fulfilled;
288	$n_{ri\varepsilon}$	maximum number of containers (TEUs) transported on leg $(r, i)$ for the OD pair $\varepsilon$ if all the demand is fulfilled;
289	$m_{risj\varepsilon}$	maximum number of containers (TEUs) transshipped from the port of call $(r, i)$ to $(s, j)$ for the OD pair $\varepsilon$ if all the demand is fulfilled; here $(r, i, s, j) \in Q$ ;
290	$N_r^{Ship}$	maximum number of ships that can be deployed on ship route $r$ ;

291	$T_{ri}^{Leg}$	minimum sailing time on leg $(r, i)$ , which is determined by ships' maximum speed;
292	$A_p^{Port}$	capacity (TEUs) of port $p$ for storing the transshipped containers;
293	$A_{ri}^{Vol}$	maximum volume capacity (in TEUs) of a ship on leg $(r, i)$ ;
294	$A_{ri}^{Load}$	maximum load capacity (in tons) of a ship on leg $(r, i)$ ;
295	$\alpha$	probability limit of overload risk for ships (e.g., 1%, 0.1%);
296	$\tilde{c}_{ri\epsilon u}$	stochastic parameter, the weight of the $u^{th}$ container on the leg $(r, i)$ for OD pair $\epsilon$ ;
297	$\mu_{ri\epsilon}$	the expected value for the random weight $\tilde{c}_{ri\epsilon u}$ ;
298	$\sigma_{ri\epsilon}$	the standard deviation for the random weight $\tilde{c}_{ri\epsilon u}$ ;
299	$C_r^{Opr}$	weekly operation cost of one ship deployed on ship route $r$ ;
300	$C^{Hold}$	unit holding cost (USD per TEU per day) of transshipped containers storing at ports;
301	$C_p^{Berth}$	penalty cost each time the dummy berth $\hat{b}$ is used at the port $p$ ;
302	$C_p^{Yard}$	penalty cost for using one TEU extra yard space for transshipped containers in port $p$ ;
303	$d_{ri}$	duration (days) of a ship dwells at the port of call $(r, i)$ ;
304	$\bar{D}$	maximum value of $d_{ri}$ for all the ports of call;
305	$l_{ri}$	length of the leg $(r, i)$ ;
306	$k_{ri}, a_{ri}$	coefficients to calculate the unit fuel cost for travelling per nautical mile on leg $(r, i)$ ;
307	$g_{bw}$	equals one if berth $b$ is available on day $w$ in a week, otherwise equals zero;

308  $f_{\dot{w}\ddot{w}w}$  equals one if day  $w$  is in time interval from day  $\dot{w}$  to  $\ddot{w}$ ; otherwise equals zero. Here  $\dot{w}, \ddot{w}, w \in W, W = \{0, 1, 2, \dots, 6\}$ . For example, if  $\dot{w} = 1, \ddot{w} = 3$ , then  $f_{\dot{w}\ddot{w}1} = f_{\dot{w}\ddot{w}2} = f_{\dot{w}\ddot{w}3} = 1$ ; if  $\dot{w} = 3, \ddot{w} = 1$ , then  $f_{\dot{w}\ddot{w}3} = f_{\dot{w}\ddot{w}4} = f_{\dot{w}\ddot{w}5} = f_{\dot{w}\ddot{w}6} = f_{\dot{w}\ddot{w}0} = f_{\dot{w}\ddot{w}1} = 1$ .

### 309 **Decision variables**

#### 310 **(1) Binary variables**

311  $\eta_{riw}$  binary variable equal to one if and only if the ship arrives at the port of call  $(r, i)$  on day  $w$  of a week;

312  $\theta_{rib}$  binary variable equal to one if and only if the ship uses berth  $b$  (including  $\hat{b}$ ) in the port of call  $(r, i)$ .

#### 313 **(2) General integer variables**

314  $\beta_r$  number of ships deployed on ship route  $r$ ; here  $\beta_r \in \{1, 2, 3, \dots, N_r^{Ship}\}$ ;

315  $\delta_{ri}$  sailing time (days) of leg  $(r, i)$ ;

316  $\tau_{ri}$  time (day) when a ship arrives at the port of call  $(r, i)$ , where  $i = 1, 2, 3, \dots, |I_r| + 1$ ;  $\tau_{r1} \in \{0, 1, 2, \dots, 6\}$ ;  $\tau_{r, |I_r|+1}$  denotes the time at which the ship completes a round trip journey;

317  $\zeta_{ri}$  auxiliary variable associated with  $\tau_{ri}$ , used to transform  $\tau_{ri}$  into a day in one week;

318  $\gamma_{risj}$  arrival time difference in days of a ship visiting  $(r, i)$  and a ship visiting  $(s, j)$ ;

319  $\xi_{risj}$  auxiliary variable associated with  $\gamma_{risj}$  to transform  $\gamma_{risj}$  into an integer less than seven;

320  $\lambda_{pw}$  extra used yard space (TEUs) for storing transshipped containers at port  $p$  on day  $w$ .

#### 321 **(3) Continuous variables**

322  $\pi_\varepsilon$  percentage of OD pair  $\varepsilon$ 's transportation demand fulfilled by the shipping liner.

323 **Mathematical model**

324 The model is then as follows:

$$\begin{aligned}
 [\mathbf{M1}] \text{ Maximize } Z = & \underbrace{\sum_{\varepsilon \in \Omega} V_{\varepsilon} \pi_{\varepsilon}}_{\text{Revenue}} - \underbrace{\sum_{r \in R} C_r^{Opr} \beta_r}_{\text{Ship operation cost}} - \underbrace{\sum_{r \in R} \sum_{i \in I_r} l_{ri} k_{ri} (l_{ri} / \delta_{ri})^{a_{ri}}}_{\text{Fuel cost}} \\
 & - \underbrace{C^{Hold} \sum_{(r,i,s,j) \in Q} \sum_{\varepsilon \in \Omega} \pi_{\varepsilon} m_{risj} \gamma_{risj}}_{\text{Holding cost of transshipment}} - \underbrace{\sum_{r \in R} \sum_{i \in I_r} C_{pri}^{Berth} \theta_{rib}}_{\text{Berth cost for extra usage}} - \underbrace{\sum_{p \in P} \sum_{w \in W} C_p^{Yard} \lambda_{pw}}_{\text{Yard cost for extra usage}}
 \end{aligned} \tag{1}$$

325

subject to

$$1 \leq \beta_r \leq N_r^{Ship} \quad r \in R \tag{2}$$

$$0 \leq \tau_{r1} \leq 6 \quad r \in R \tag{3}$$

$$\delta_{ri} \geq T_{ri}^{Leg} \quad r \in R, i \in I_r \tag{4}$$

$$\tau_{r,i+1} = \tau_{ri} + d_{ri} + \delta_{ri} \quad r \in R, i \in I_r \tag{5}$$

$$\tau_{r,|I_r|+1} = \tau_{r1} + 7\beta_r \quad r \in R \tag{6}$$

$$\sum_{w \in W} \eta_{riw} = 1 \quad r \in R, i \in I_r \tag{7}$$

$$\tau_{ri} = \sum_{w \in W} w \eta_{riw} + 7\zeta_{ri} \quad r \in R, i \in I_r \tag{8}$$

$$0 \leq \zeta_{ri} \leq \beta_r - 1 \quad r \in R, i \in I_r \tag{9}$$

$$\tau_{sj} - \tau_{ri} + 7\xi_{risj} = \gamma_{risj} \quad (r, i, s, j) \in Q \tag{10}$$

$$0 \leq \gamma_{risj} \leq 6 \quad (r, i, s, j) \in Q \tag{11}$$

$$\sum_{b \in B_{pri} \cup \{\hat{b}\}} \theta_{rib} = 1 \quad r \in R, i \in I_r \tag{12}$$

$$\sum_{r \in R'_p} \sum_{v=1}^{\bar{D}} \sum_{i \in I'_{rp}: d_{ri}=v} \sum_{k=0}^{v-1} \theta_{rib} \eta_{r,i,(w-k+7) \bmod 7} \leq g_{bw} \quad p \in P, b \in B_p, w \in W \tag{13}$$

$$\left( \sum_{(r,i,s,j) \in Q_p} \sum_{\varepsilon \in \Omega} \pi_\varepsilon m_{risj\varepsilon} \sum_{\dot{w}, \ddot{w} \in W} \eta_{ri\dot{w}} \eta_{sj\ddot{w}} f_{\dot{w}\ddot{w}w} - A_p^{Port} \right)^+ = \lambda_{pw} \quad p \in P, w \in W \quad (14)$$

$$\sum_{\varepsilon \in \Omega} \pi_\varepsilon n_{ri\varepsilon} \leq A_{ri}^{Vol} \quad r \in R, i \in I_r \quad (15)$$

$$Prob\left(\sum_{\varepsilon \in \Omega} \sum_{u=1}^{\lceil \pi_\varepsilon n_{ri\varepsilon} \rceil} \tilde{c}_{ri\varepsilon u} > A_{ri}^{Load}\right) \leq \alpha \quad r \in R, i \in I_r \quad (16)$$

$$0 \leq \pi_\varepsilon \leq 1 \quad \varepsilon \in \Omega \quad (17)$$

$$\beta_r \in \mathbb{Z}_+ \quad r \in R \quad (18)$$

$$\tau_{ri} \in \mathbb{Z}_+ \cup \{0\} \quad r \in R, i \in I_r \cup \{|I_r| + 1\} \quad (19)$$

$$\eta_{riw} \in \{0, 1\} \quad r \in R, i \in I_r, w \in W \quad (20)$$

$$\delta_{ri} \in \mathbb{Z}_+ \cup \{0\} \quad r \in R, i \in I_r \quad (21)$$

$$\zeta_{ri} \in \mathbb{Z}_+ \cup \{0\} \quad r \in R, i \in I_r \quad (22)$$

$$\gamma_{risj} \in \mathbb{Z}_+ \cup \{0\} \quad (r, i, s, j) \in Q \quad (23)$$

$$\xi_{risj} \in \mathbb{Z} \quad (r, i, s, j) \in Q \quad (24)$$

$$\theta_{rib} \in \{0, 1\} \quad r \in R, i \in I_r, b \in B_{p_{ri}} \cup \{\hat{b}\} \quad (25)$$

$$\lambda_{pw} \geq 0 \quad p \in P, w \in W. \quad (26)$$

326 The objective (1) is to maximize the revenue, minus the five types of cost de-  
 327 scribed in Section 3. Constraints (2) state that at least one ship and at most  $N_r^{Ship}$   
 328 ships should be deployed on each route. Constraints (3) ensure the start time of each  
 329 route (service) occurs in the first week. Constraints (4) relate to the minimum re-  
 330 quired sailing time  $T_{ri}^{Leg}$  for each leg, which depends on the maximum speed of ships.  
 331 Constraints (5) link the arrival time  $\tau_{ri}$  of a port of call with the arrival time  $\tau_{r,i+1}$   
 332 of the next port of call on a route. Constraints (6) guarantee that the total number  
 333 of days  $\tau_{r,|I_r|+1} - \tau_{r1}$  for a ship completing its travel on a route is the number of  
 334 ships deployed on the route times seven, because all the services follow weekly arrival  
 335 pattern and one week has seven days. Constraints (7)–(9) link the binary variable  
 336  $\eta_{riw}$  and the integer variable  $\tau_{ri}$ , both of which denote the arrival time of the  $i^{th}$



337 port of call on ship route  $r$ . The difference is that  $\tau_{ri}$  denotes the arrival time on a  
 338 universal time axis, while  $\mu_{ri}^w$  denotes the arrival time in one of the seven days in a  
 339 week. The former is from the perspective of port arrival time in one ship's itinerary  
 340 (e.g., day 2 at port 1, day 11 at port 2), while the latter is from the perspective of  
 341 the port arrival time of a fleet of ships deployed on a route (e.g., Mon at port 1,  
 342 Wed at port 2). Constraints (10)–(11) transfer the absolute time gap (days)  $\tau_{sj} - \tau_{ri}$   
 343 between two ports of call  $(r, i)$  and  $(s, j)$  to a time difference  $\gamma_{risj}$  in days within one  
 344 week. Similarly, the former is from the perspective of port arrival time in two ship's  
 345 itineraries for two routes (e.g., a ship in route 1 arrives at port  $p$  on day 2, a ship  
 346 in route 2 arrives at the port  $p$  on day 11, and the absolute time difference is nine  
 347 days), while the latter is from the perspective of the port arrival time of two fleets of  
 348 ships deployed on two routes (e.g., route 1's fleet arrives at the port on Mon, route  
 349 2's fleet arrives at the port on Wed, and the time difference is two days, which is the  
 350 waiting time for transshipment from route 1 to route 2). Constraints (12) guarantee  
 351 that each port of call of a route should be assigned a berth (one of reserved berths or  
 352 the dummy berth  $\hat{b}$ ). The berth availability limitation is ensured by Constraints (13),  
 353 which are not straightforward and will be explained later. Constraints (14) calculate  
 354 the extra used yard space (TEUs) for storing transshipped containers at each port on  
 355 each day. Constraints (15) define the limitation of the ship capacity with respect to  
 356 its available space during each leg. Constraints (16) mean that the overload probabili-  
 357 ty is lower than a threshold  $\alpha$ . Constraints (17)–(26) state the ranges of the defined  
 358 decision variables.

359 More explanations are required for Constraints (13). In the simplest case where  
 360 all ships dwell at ports for only one day, the left-hand side of the constraint is  
 361  $\sum_{r \in R_p'} \sum_{i \in I'_{rp}} \theta_{rib} \eta_{riw}$ , which denotes whether or not one of the reserved berths  $b$   
 362 is used by a ship on day  $w$  in a week. This value should not be greater than  $g_{bw}$ ,  
 363 which is the availability of the berth. If some ships dwell at a port for one day (i.e.,  
 364  $d_{ri} = 1$ ), and some ships dwell for two days (i.e.,  $d_{ri} = 2$ ), the calculation on whether  
 365 or not berth  $b$  is used by the  $i^{th}$  port of call on ship route  $r$  is as follows: (1) if  $w$   
 366  $= 1, 2, 3, \dots, 6$ , then  $\sum_{r \in R_p'} [\sum_{i \in I'_{rp}: d_{ri}=1} \theta_{rib} \eta_{riw} + \sum_{i \in I'_{rp}: d_{ri}=2} (\theta_{rib} \eta_{r, i, w-1} + \theta_{rib} \eta_{riw})]$ ;  
 367 (2) if  $w = 0$ , then  $\sum_{r \in R_p'} [\sum_{i \in I'_{rp}: d_{ri}=1} \theta_{rib} \eta_{riw} + \sum_{i \in I'_{rp}: d_{ri}=2} (\theta_{rib} \eta_{r, i, w-1+7} + \theta_{rib} \eta_{riw})]$ .  
 368 In what follows, subscripts  $w - 1$  and  $w - 1 + 7$  are interpreted as  $(w - 1 + 7) \bmod$   
 369 7. Then suppose the ships' dwell time can be one, two,  $\dots$ , or at most  $\bar{D}$  days, then

the above formula becomes  $\sum_{r \in R_p'} \sum_{v=1}^{\bar{D}} \sum_{i \in I'_{rp}: d_{ri}=v} \sum_{k=0}^{v-1} (\theta_{rib} \eta_{r,i,(w-k+7) \bmod 7})$ . This value does not exceed  $g_{bw}$  by Constraints (13).

## 5. Linearization of the model

The above model **[M1]** is an optimization problem with integer decision variables and non-linear terms that are non-convex. It is difficult to solve it using off-the-shelf solvers because (i) it contains a large number of discrete variables and (ii) it has a non-linear objective function and non-linear constraints. To solve this model, we first linearize it, and we then develop a sequential optimization algorithm.

### 5.1. Linearization of Objective (1)

Objective (1) contains a non-linear part  $\sum_{r \in R} \sum_{i \in I_r} l_{ri} k_{ri} (l_{ri} / \delta_{ri})^{a_{ri}}$ , which can be rewritten as  $\sum_{r \in R} \sum_{i \in I_r} l_{ri} k_{ri} l_{ri}^{a_{ri}} \delta_{ri}^{-a_{ri}}$ . The key is to transform  $\delta_{ri}^{-a_{ri}}$  into a linear form. We adopt the linearization method used by Wang et al. (2013). We first redefine  $\delta_{ri}$  as a new binary variable  $\delta'_{rit}$ , which denotes whether or not the sailing time for the  $i^{th}$  leg of ship route  $r$  equals  $t$  days,  $t \in T$ , where  $T$  is the set of integers denoting the possible sailing times (in days) for all legs; for example  $T \in \{1, \dots, 15\}$ . The non-linear form  $\delta_{ri}^{-a_{ri}}$  can then be replaced with  $\sum_{t \in T} \delta'_{rit} t^{-a_{ri}}$ , subject to  $\sum_{t \in T} \delta'_{rit} = 1$  for all  $r \in R, i \in I_r$ .

Objective (1) contains another non-linear part  $\sum_{(r,i,s,j) \in Q} \pi_\varepsilon m_{risj} \gamma_{risj}$ , which can be linearized as follows Alharbi et al. (2015). We first transform the integer variable  $\gamma_{risj}$  into a binary variable. Since  $\gamma_{risj} \in W$ , we redefine  $\gamma_{risj}$  as a binary variable  $\gamma'_{risjw}$ , equal to one if and only if the time gap between ports of call  $(r, i)$  and  $(s, j)$  is  $w$  days. Then  $\gamma_{risj}$  is replaced with  $\sum_{w \in W} w \gamma'_{risjw}$ , subject to  $\sum_{w \in W} \gamma'_{risjw} = 1$  for all  $(r, i, s, j) \in Q$ . Here both  $\pi_\varepsilon$  and  $\gamma'_{risjw}$  are binary variables; therefore, the value of  $M$  is 1.

394

Based on the above linearization, Objective (1) becomes

$$\begin{aligned}
\text{Maximize } Z = & \underbrace{\sum_{\varepsilon \in \Omega} V_{\varepsilon} \pi_{\varepsilon}}_{\text{Revenue}} - \underbrace{\sum_{r \in R} C_r^{Opr} \beta_r}_{\text{Ship operation cost}} - \underbrace{\sum_{r \in R} \sum_{i \in I_r} l_{ri} k_{ri} l_{ri}^{a_{ri}} \sum_{t \in T} \delta'_{rit} t^{-a_{ri}}}_{\text{Fuel cost}} \\
& - \underbrace{C^{Hold} \sum_{(r,i,s,j) \in Q} \sum_{w \in W} m_{risj\varepsilon} w \varrho_{risjw\varepsilon}}_{\text{Holding cost of transshipment}} - \underbrace{\sum_{r \in R} \sum_{i \in I_r} C_{pri}^{Berth} \theta_{rib}}_{\text{Berth cost for extra usage}} - \underbrace{\sum_{p \in P, w \in W} C_p^{Yard} \lambda_{pw}}_{\text{Yard cost for extra usage}}
\end{aligned} \tag{27}$$

395

The newly defined variables and constraints needed for this linearization are sum-

396

marized as follows:

397

**Newly defined indices, sets and parameters:**

398

$t$  index of the number of days;

399

$T$  set of possible numbers of days for a leg's sailing time,  $T = \{1, \dots, |T|\}$ ;

400

$M$  a sufficiently large positive number.

401

**Newly defined variables:**

402

$\delta'_{rit}$  a binary variable equal to one if and only if the sailing time of the leg  $(r, i)$  is  $t$ ;

403

$\gamma'_{risjw}$  a binary variable equal to one if and only if the time gap between the ports of call  $(r, i)$  and  $(s, j)$  (i.e.,  $\gamma_{risj}$ ) is  $w$  days;

404

$\varrho_{risjw\varepsilon}$  continuous variable equal to  $\pi_{\varepsilon} \gamma'_{risjw}$  if  $\gamma'_{risjw} = 1$ ; otherwise zero.

405

**Newly defined constraints:**

Constraints (11) are removed. Constraints (5), (10), (21), (23) are replaced with the following four constraints, respectively.

$$\tau_{r,i+1} = \tau_{ri} + d_{ri} + \sum_{t \in T} t \delta'_{rit} \quad r \in R, i \in I_r \tag{28}$$

$$\tau_{sj} - \tau_{ri} + 7\xi_{risj} = \sum_{w \in W} w \gamma'_{risjw} \quad (r, i, s, j) \in Q \tag{29}$$

$$\delta'_{rit} \in \{0, 1\} \quad r \in R, i \in I_r, t \in T \tag{30}$$

$$\gamma'_{risjw} \in \{0, 1\} \quad (r, i, s, j) \in Q, w \in W. \quad (31)$$

In addition, four new constraints are defined:

$$\sum_{t \in T} \delta'_{rit} = 1 \quad r \in R, i \in I_r \quad (32)$$

$$\sum_{w \in W} \gamma'_{risjw} = 1 \quad (r, i, s, j) \in Q \quad (33)$$

$$0 \leq \varrho_{risjw\varepsilon} \leq 1 \quad (r, i, s, j) \in Q, w \in W, \varepsilon \in \Omega. \quad (34)$$

## 406 5.2. Linearization of Constraints (13)

407 Constraints (13) contain a non-linear part  $\theta_{rib}\eta_{r,i,(w-k+7) \bmod 7}$ , which is the prod-  
408 uct of two binary variables. Following the method used by Yi et al. (2018), we define  
409 a new binary variable  $\varphi_{ribw}$  to replace the non-linear part.

### 410 Newly defined variables:

411  $\varphi_{ribw}$  binary variable equal to one if and only if the ship arrives at the berth  $b$   
on the day  $w$  of a week in the  $i^{th}$  port of call on ship route  $r$ .

Then Constraints (13) become

$$\sum_{r \in R'_p} \sum_{v=1}^{\bar{D}} \sum_{i \in I'_{rp}: d_{ri}=v} \sum_{k=0}^{v-1} \theta_{r,i,b,(w-k+7) \bmod 7} \leq g_{bw} \quad p \in P, b \in B_p, w \in W. \quad (35)$$

In addition, some more constraints need to be defined so that the newly defined variable  $\varphi_{ribw}$  can replace the function of  $\theta_{rib}\eta_{r,i,(w-k+7) \bmod 7}$ .

$$\varphi_{ribw} \geq \theta_{rib} + \eta_{riw} - 1 \quad r \in R, i \in I_r, b \in B_{p_{ri}}, w \in W \quad (36)$$

$$\varphi_{ribw} \leq \theta_{rib} \quad r \in R, i \in I_r, b \in B_{p_{ri}}, w \in W \quad (37)$$

$$\varphi_{ribw} \leq \eta_{riw} \quad r \in R, i \in I_r, b \in B_{p_{ri}}, w \in W \quad (38)$$

$$\varphi_{ribw} \in \{0, 1\} \quad r \in R, i \in I_r, b \in B_{p_{ri}}, w \in W. \quad (39)$$

## 412 5.3. Linearization of Constraints (14)

413 Constraints (14) contain the product of three variables  $\pi_\varepsilon, \eta_{riw}$  and  $\eta_{sj\ddot{w}}$ . In ad-  
414 dition, the form  $\lambda_{pw} = (\cdot)^+$  is also non-linear. In the first case, we use an approach

similar to that of Section 5.2 to handle it. This approach was used by Wang and Meng (2012). We define some more decision variables and constraints:

**Newly defined variables:**

$\psi_{risj\dot{w}\ddot{w}}$  binary variable equal to one if and only if both variables  $\eta_{ri\dot{w}}$  and  $\eta_{sj\ddot{w}}$  are equal to one;

$\phi_{risj\dot{w}\ddot{w}\varepsilon}$  binary variable equal to  $\pi_\varepsilon$  if and only if  $\psi_{risj\dot{w}\ddot{w}} = 1$ .

Then Constraints (14) become

$$\lambda_{pw} = \left( \sum_{(r,i,s,j) \in Q_p} \sum_{\varepsilon \in \Omega} \sum_{\dot{w}, \ddot{w} \in W} m_{risj\varepsilon} f_{\dot{w}\ddot{w}} \phi_{risj\dot{w}\ddot{w}\varepsilon} - A_p^{Port} \right)^+ \quad p \in P, w \in W. \quad (40)$$

In addition, some more constraints need to be defined as follows so that the newly defined variable  $\psi_{risj\dot{w}\ddot{w}}$  can replace the function of  $\eta_{ri\dot{w}}\eta_{sj\ddot{w}}$ :

$$\psi_{risj\dot{w}\ddot{w}} \geq \eta_{ri\dot{w}} + \eta_{sj\ddot{w}} - 1 \quad (r, i, s, j) \in Q; \dot{w}, \ddot{w} \in W \quad (41)$$

$$\psi_{risj\dot{w}\ddot{w}} \leq \eta_{ri\dot{w}} \quad (r, i, s, j) \in Q; \dot{w}, \ddot{w} \in W \quad (42)$$

$$\psi_{risj\dot{w}\ddot{w}} \leq \eta_{sj\ddot{w}} \quad (r, i, s, j) \in Q; \dot{w}, \ddot{w} \in W \quad (43)$$

$$\psi_{risj\dot{w}\ddot{w}} \in \{0, 1\} \quad (r, i, s, j) \in Q; \dot{w}, \ddot{w} \in W \quad (44)$$

$$\phi_{risj\dot{w}\ddot{w}\varepsilon} \geq \pi_\varepsilon + (\psi_{risj\dot{w}\ddot{w}} - 1)M \quad (r, i, s, j) \in Q; \dot{w}, \ddot{w} \in W, \varepsilon \in \Omega \quad (45)$$

$$0 \leq \phi_{risj\dot{w}\ddot{w}\varepsilon} \leq 1 \quad (r, i, s, j) \in Q; \dot{w}, \ddot{w} \in W, \varepsilon \in \Omega. \quad (46)$$

For the non-linear part  $\lambda_{pw} = (\cdot)^+$ , we adopt the linearization method used by Wang and Meng (2015). We define two more non-negative variables  $\lambda_{pw}^+$  and  $\lambda_{pw}^-$ , and Constraints (40) are changed into

$$\sum_{(r,i,s,j) \in Q_p} \sum_{\varepsilon \in \Omega} \sum_{\dot{w}, \ddot{w} \in W} m_{risj\varepsilon} f_{\dot{w}\ddot{w}} \phi_{risj\dot{w}\ddot{w}\varepsilon} - A_p^{Port} = \lambda_{pw}^+ - \lambda_{pw}^- \quad p \in P, w \in W. \quad (47)$$

Then Constraints (26) are replaced with

$$\lambda_{pw}^+, \lambda_{pw}^- \geq 0 \quad p \in P, w \in W. \quad (48)$$

Moreover, Objective (27) is further restated by replacing  $\lambda_{pw}$  with  $\lambda_{pw}^+$ . Then the final version of the objective becomes

$$\begin{aligned}
\text{Maximize } Z = & \underbrace{\sum_{\varepsilon \in \Omega} V_{\varepsilon} \pi_{\varepsilon}}_{\text{Revenue}} - \underbrace{\sum_{r \in R} C_r^{Opr} \beta_r}_{\text{Ship operation cost}} - \underbrace{\sum_{r \in R} \sum_{i \in I_r} l_{ri} k_{ri} l_{ri}^{a_{ri}} \sum_{t \in T} \delta'_{rit} t^{-a_{ri}}}_{\text{Fuel cost}} \\
& - \underbrace{C^{Hold} \sum_{\langle r, i, s, j \rangle \in Q} \sum_{w \in W} m_{risj\varepsilon} w Q_{risjw\varepsilon}}_{\text{Holding cost of transshipment}} - \underbrace{\sum_{r \in R} \sum_{i \in I_r} C_{pri}^{Berth} \theta_{ri\hat{b}}}_{\text{Berth cost for extra usage}} - \underbrace{\sum_{p \in P, w \in W} C_p^{Yard} \lambda_{pw}^+}_{\text{Yard cost for extra usage}}.
\end{aligned} \tag{49}$$

**Lemma 1.** *Because the weights of the  $\lceil \pi_{\varepsilon} n_{ri\varepsilon} \rceil$  containers are independent and identically distributed random variables with expected values  $u_{ri\varepsilon}$  and variances  $\sigma_{ri\varepsilon}^2$ , the classical central limit theorem (CLT) states that since  $\lceil \pi_{\varepsilon} n_{ri\varepsilon} \rceil$  is very large, the distribution of the total weight  $\sum_{u=1}^{\lceil \pi_{\varepsilon} n_{ri\varepsilon} \rceil} \tilde{c}_{ri\varepsilon u}$  is approximately normal with mean  $\lceil \pi_{\varepsilon} n_{ri\varepsilon} \rceil \mu_{ri\varepsilon}$  and variance  $\lceil \pi_{\varepsilon} n_{ri\varepsilon} \rceil \sigma_{ri\varepsilon}^2$ .*

**Lemma 2.** *When  $\lceil \pi_{\varepsilon} n_{ri\varepsilon} \rceil$  is very large,  $r \in R, i \in I_r, \varepsilon \in \Omega$ , since the containers weights are independent, the total weight  $\sum_{\varepsilon \in \Omega} \sum_{u=1}^{\lceil \pi_{\varepsilon} n_{ri\varepsilon} \rceil} \tilde{c}_{ri\varepsilon u}$  of all the carried containers approximately follows a normal distribution  $N(\sum_{\varepsilon \in \Omega} \lceil \pi_{\varepsilon} n_{ri\varepsilon} \rceil \mu_{ri\varepsilon}, \sum_{\varepsilon \in \Omega} \lceil \pi_{\varepsilon} n_{ri\varepsilon} \rceil \sigma_{ri\varepsilon}^2)$ .*

In reality, the number of containers is large. According to Lemma (2), Constraints (16) can be approximated by

$$\sum_{\varepsilon \in \Omega} \lceil \pi_{\varepsilon} n_{ri\varepsilon} \rceil \mu_{ri\varepsilon} + z_{1-\alpha} \left( \sum_{\varepsilon \in \Omega} \lceil \pi_{\varepsilon} n_{ri\varepsilon} \rceil \sigma_{ri\varepsilon}^2 \right)^{\frac{1}{2}} \leq A_{ri}^{Load} \quad r \in R, i \in I_r, \tag{50}$$

where  $z_{1-\alpha}$  is the 100(1 -  $\alpha$ ) percentile of the standard normal distribution.

**Proposition 1.** *The left-hand sides of Constraints (50) are in general non-convex in  $\pi_{\varepsilon}$ .*

*Proof.* To prove the proposition, we just need to provide a non-convex example. Consider a simple case with only one OD pair, i.e.,  $|\Omega| = 1$ . Suppose that for this OD pair  $\varepsilon$ , we have  $\mu = 1, \sigma^2 = 0.25$ . Suppose further than  $z = 1$  and  $n_{ri\varepsilon} = 10$ . Then the left-hand side of the constraint becomes  $10\pi + 0.25\sqrt{10\pi}$ . Consider three values of

437  $\pi$ :  $\pi_1 = 0$ ,  $\pi_2 = 1$ , and  $\pi_3 = 2$ . Then  $10\pi_1 + 0.25\sqrt{10\pi_1} = 0$ ,  $10\pi_2 + 0.25\sqrt{10\pi_2} = 1.25$ ,  
 438  $10\pi_3 + 0.25\sqrt{10\pi_3} = 2.35$ . In other words,  $\pi_2 = (\pi_1 + \pi_3)/2 = (0 + 2)/2 = 1$ , however,  
 439  $10\pi_2 + 0.25\sqrt{10\pi_2} > (10\pi_1 + 0.25\sqrt{10\pi_1} + 10\pi_3 + 0.25\sqrt{10\pi_3})/2$ . Therefore, the left-  
 440 hand side of the constraint in this case is non-convex.  $\square$

441 In order to handle the non-convex Constraints (50), we propose a second-order  
 442 cone programming (SOCP)-based algorithm, which will be elaborated in Section 6.

## 443 6. Algorithmic strategy

444 We now present an SOCP-based algorithm to handle non-convex constraints in the  
 445 model. A dynamic linearization algorithm and a tabu search algorithm are applied  
 446 to solve the model under different scales of route networks.

### 447 6.1. SOCP transformation

We use SOCP to transfer Constraints (50) to a convex one. We first define a new  
 binary variable  $\kappa_{ri\varepsilon h}$  to represent the integer  $\lceil \pi_{\varepsilon} n_{ri\varepsilon} \rceil$ :

$$\lceil \pi_{\varepsilon} n_{ri\varepsilon} \rceil = \sum_{h=0}^{H_{ri\varepsilon}} 2^h \kappa_{ri\varepsilon h} \quad r \in R, i \in I_r, \varepsilon \in \Omega, h = 0, 1, \dots, H_{ri\varepsilon} \quad (51)$$

$$\sum_{h=0}^{H_{ri\varepsilon}} 2^h \kappa_{ri\varepsilon h} \leq n_{ri\varepsilon} \quad r \in R, i \in I_r, \varepsilon \in \Omega, h = 0, 1, \dots, H_{ri\varepsilon} \quad (52)$$

$$\kappa_{ri\varepsilon h} \in \{0, 1\} \quad r \in R, i \in I_r, \varepsilon \in \Omega, h = 0, 1, \dots, H_{ri\varepsilon}, \quad (53)$$

where  $H_{ri\varepsilon} := \lceil \log_2 n_{ri\varepsilon} \rceil$ . Then Constraints (50) become

$$\sum_{\varepsilon \in \Omega} \mu_{ri\varepsilon} \sum_{h=0}^{H_{ri\varepsilon}} 2^h \kappa_{ri\varepsilon h} + z_{1-\alpha} \left( \sum_{\varepsilon \in \Omega} \sigma_{ri\varepsilon}^2 \sum_{h=0}^{H_{ri\varepsilon}} 2^h \kappa_{ri\varepsilon h} \right)^{\frac{1}{2}} \leq A_{ri}^{Load} \quad r \in R, i \in I_r. \quad (54)$$

Since  $\kappa_{ri\varepsilon h}$  is binary, we have  $\kappa_{ri\varepsilon h} = \kappa_{ri\varepsilon h}^2$ . Using this property, Constraints (54)  
 become

$$\left( \sum_{\varepsilon \in \Omega} \sum_{h=0}^{H_{ri\varepsilon}} 2^h \sigma_{ri\varepsilon}^2 \kappa_{ri\varepsilon h}^2 \right)^{\frac{1}{2}} \leq (A_{ri}^{Load} - \sum_{\varepsilon \in \Omega} \sum_{h=0}^{H_{ri\varepsilon}} 2^h \mu_{ri\varepsilon} \kappa_{ri\varepsilon h}) / z_{1-\alpha} \quad r \in R, i \in I_r. \quad (55)$$

Constraints (55) are convex and now the following  $[M2]$  is a mixed integer SOCP (MISOCP) model, which can be solved by off-the-shelf solvers such as CPLEX.

$[M2]$  An MISOCP model: Objective (49)

subject to Constraints (2)–(4), (6)–(9), (11)–(12), (15)–(20), (22), (24)–(25), (28)–(39), (41)–(48), (52)–(53), (55).

### 6.2. Dynamic linearization for solving $[M2]$

We propose solving the MISOCP model  $[M2]$  by integer linear programming. The core idea is as follows: since Constraints (55) are convex, if we know an infeasible solution  $\check{\mathbf{y}} := (\check{\kappa}_{ri\varepsilon h}, r \in R, i \in I_r, \varepsilon \in \Omega, h = 0, 1, 2, \dots, H_{ri\varepsilon})$  that violates the non-linear Constraints (55), we can linearize the left-hand side  $(\sum_{\varepsilon \in \Omega} \sum_{h=0}^{H_{ri\varepsilon}} 2^h \sigma_{ri\varepsilon}^2 \kappa_{ri\varepsilon h}^2)^{\frac{1}{2}}$  of the constraint at  $\check{\mathbf{y}}$ . Note that  $\frac{\partial (\sum_{\varepsilon \in \Omega} \sum_{h=0}^{H_{ri\varepsilon}} 2^h \sigma_{ri\varepsilon}^2 \kappa_{ri\varepsilon h}^2)^{\frac{1}{2}}}{\partial \kappa_{ri\varepsilon h}} = \frac{2^h \sigma_{ri\varepsilon}^2 \check{\kappa}_{ri\varepsilon h}}{(\sum_{\varepsilon \in \Omega} \sum_{h=0}^{H_{ri\varepsilon}} 2^h \sigma_{ri\varepsilon}^2 \check{\kappa}_{ri\varepsilon h}^2)^{\frac{1}{2}}}$  at  $\check{\mathbf{y}}$ . Hence, we can add the resulting linear constraint to the model in order to cut off the infeasible solution  $\check{\mathbf{y}}$ , as well as some other infeasible solutions. We propose the following Algorithm 1 to solve model  $[M2]$  and we then prove its correctness.

---

**Algorithm 1** Dynamic linearization algorithm for solving  $[M2]$

---

**Step 1.** Define a set  $\Psi$  of generated intermediate infeasible solutions of  $\mathbf{y} := (\kappa_{ri\varepsilon h}, r \in R, i \in I_r, \varepsilon \in \Omega, h = 0, 1, 2, \dots, H_{ri\varepsilon})$ . Initialize  $\Psi \leftarrow \emptyset$ .

**Step 2.** Solve model  $[M3]$  whose objective function is Eq. (49) subject to Constraints (2)–(4), (6)–(9), (11)–(12), (15)–(20), (22), (24)–(25), (28)–(39), (41)–(48), (52)–(53) and the following constraints:

$$\frac{\sum_{\varepsilon \in \Omega} \sum_{h=0}^{H_{ri\varepsilon}} 2^h \sigma_{ri\varepsilon}^2 \check{\kappa}_{ri\varepsilon h} (\kappa_{ri\varepsilon h} - \check{\kappa}_{ri\varepsilon h})}{(\sum_{\varepsilon \in \Omega} \sum_{h=0}^{H_{ri\varepsilon}} 2^h \sigma_{ri\varepsilon}^2 \check{\kappa}_{ri\varepsilon h}^2)^{\frac{1}{2}}} + (\sum_{\varepsilon \in \Omega} \sum_{h=0}^{H_{ri\varepsilon}} 2^h \sigma_{ri\varepsilon}^2 \check{\kappa}_{ri\varepsilon h}^2)^{\frac{1}{2}} \leq \frac{A_{ri}^{Load} - \sum_{\varepsilon \in \Omega} \sum_{h=0}^{H_{ri\varepsilon}} 2^h \mu_{ri\varepsilon} \kappa_{ri\varepsilon h}}{z_{1-\alpha}}, \check{\mathbf{y}} \in \Psi, r \in R, i \in I_r. \quad (56)$$

Let  $\hat{\mathbf{y}}$  be the optimal solution to model  $[M3]$ .

**Step 3.** Check whether  $\hat{\mathbf{y}}$  satisfies Constraints (55). If yes, then  $\hat{\mathbf{y}}$  is the optimal solution to  $[M2]$  and stop. Otherwise, set  $\Psi \leftarrow \Psi \cup \{\hat{\mathbf{y}}\}$  and go to **Step 1**.

---



462 **Proposition 2.** *No solution will be generated twice in Algorithm 1.*

463 *Proof.* If a generated solution  $\hat{\mathbf{y}}$  is feasible with respect to Constraints (55), then the  
 464 algorithm stops and hence it will not be generated twice. If it is infeasible, then it  
 465 will become an element of  $\Psi$  at the next iteration and we denote it by  $\check{\mathbf{y}}$ . Since  $\check{\mathbf{y}}$  is  
 466 infeasible, we have

$$\left( \sum_{\varepsilon \in \Omega} \sum_{h=0}^{H_{ri\varepsilon}} 2^h \sigma_{ri\varepsilon}^2 \check{\kappa}_{ri\varepsilon h}^2 \right)^{\frac{1}{2}} > \frac{A_{ri}^{Load} - \sum_{\varepsilon \in \Omega} \sum_{h=0}^{H_{ri\varepsilon}} 2^h \mu_{ri\varepsilon} \check{\kappa}_{ri\varepsilon h}}{z_{1-\alpha}} \quad r \in R, i \in I_r. \quad (57)$$

467 Inequality (57) implies that  $\hat{\mathbf{y}} = \check{\mathbf{y}}$  violates the added Constraints (56). Hence,  $\hat{\mathbf{y}} = \check{\mathbf{y}}$   
 468 will not be generated again.  $\square$

469 **Proposition 3.** *Algorithm 1 terminates in a finite number of iterations.*

470 *Proof.* Since all  $\kappa_{ri\varepsilon h}$  variables are binary for  $r \in R, i \in I_r, \varepsilon \in \Omega$ , and  $h = 0, 1, \dots$   
 471  $\cdot, H_{ri\varepsilon}$ , the number of solutions feasible to Constraints (2)–(4), (6)–(9), (11)–(12),  
 472 (15)–(20), (22), (24)–(25), (28)–(39), (41)–(48) and (52)–(53) is at most  $2^{\sum_{r \in R} \sum_{i \in I_r} \sum_{\varepsilon \in \Omega} (1+H_{ri\varepsilon})}$ .  
 473 Proposition 2 implies that at least one solution is excluded at each iteration. Hence,  
 474 Algorithm 1 terminates in at most  $2^{\sum_{r \in R} \sum_{i \in I_r} \sum_{\varepsilon \in \Omega} (1+H_{ri\varepsilon})}$  iterations.  $\square$

475 **Proposition 4.** *An optimal solution is obtained when Algorithm 1 terminates.*

476 *Proof.* Model  $[\mathbf{M3}]$  is a relaxation of the original model  $[\mathbf{M2}]$ , because the lin-  
 477 earization on the left-hand side of inequality (56) underestimates the convex function  
 478  $\left( \sum_{\varepsilon \in \Omega} \sum_{h=0}^{H_{ri\varepsilon}} 2^h \sigma_{ri\varepsilon}^2 \kappa_{ri\varepsilon h}^2 \right)^{\frac{1}{2}}$ . Since  $[\mathbf{M2}]$  and  $[\mathbf{M3}]$  have the same objective function, the  
 479 value of  $\hat{\mathbf{y}}$  generated in **Step 1** is at least equal to that of the optimal value of  $[\mathbf{M2}]$ .  
 480 If  $\hat{\mathbf{y}}$  is feasible for  $[\mathbf{M2}]$ , then the objective value of the feasible solution  $\hat{\mathbf{y}}$  to  $[\mathbf{M2}]$   
 481 is equal to an upper bound (the optimal objective value of  $[\mathbf{M3}]$ ), meaning that  $\hat{\mathbf{y}}$   
 482 is optimal for  $[\mathbf{M2}]$ .  $\square$

### 483 6.3. Tabu search algorithm for solving $[\mathbf{M2}]$

484 We now propose a tabu search algorithm to solve  $[\mathbf{M2}]$ . Tabu search algorithm,  
 485 introduced by Glover (1986), is an adaptive local iterative search that operates within

486 a solution space. It moves from one solution to another and diversifies solutions  
487 so as to find a better one (Vivaldini et al., 2016). At each iteration, the search  
488 process is applied to explore the neighborhood of the current optimal solution. Tabu  
489 search algorithm has often been applied to problems solving in the maritime industry.  
490 Cordeau et al. (2005) applied a tabu search algorithm to the berth allocation problem  
491 (BAP). Tirado et al. (2013) solved a dynamic and stochastic cargo transportation  
492 problem by means of tabu search. Nikolopoulou et al. (2017) used tabu search to  
493 compare two kinds of cargo transportation methods in the shipping industry.

#### 494 *6.3.1. Local optimization using tabu search*

495 Given a neighborhood structure ( $N(p_c)$ ) and an initial solution  $p$ , the tabu search  
496 algorithm iteratively replaces the incumbent solution  $p_c$  by a best eligible neighbor  
497 solution ( $\hat{p} \in N(p_c)$ ) until a stopping criterion is met, i.e., the current optimal solution  
498  $p^*$  has not been improved for  $T_{max}$  consecutive iterations. At each iteration, the best  
499 movement is recorded in the tabu list to prevent the reverse movement in the next  
500 iterations. A movement is eligible if it is not in the tabu list or if it results in a better  
501 solution than the current optimal solution. The general tabu search framework is  
502 described in Algorithm 2 and the details are explained in subsequent sections.

#### 503 *6.3.2. Population initialization*

504 The population initialization is obtained by generating 10 random solutions using  
505 a uniform probability distribution. The component  $x_{r,i}$  of each solution is randomly  
506 assigned a value from  $[T_{ri}^{min}, T_{ri}^{max}]$ , where  $r \in R, i \in I_r \setminus \{|I_r|\}$ . The minimum value  
507  $T_{ri}^{min}$  and maximum value  $T_{ri}^{max}$  refer to the minimum sailing time and the maximum  
508 sailing time of each leg of each route according to the maximum speed and minimum  
509 sailing speed, respectively. Moreover, we should guarantee that the sum of the sailing  
510 time on each leg plus the duration time at each port is the multiple of seven by  
511 adjusting the time of the last component  $x_{r,|I_r|}$ , where  $r \in R$ . We then select the best  
512 solution  $p_0$  among the 10 random solutions as the initial solution  $p$ .

#### 513 *6.3.3. Neighborhood structure and movement*

514 The neighborhood structure is the crucial component of the algorithm. The neigh-  
515 borhood  $N(p_c)$  contains all solutions in which the value of one component is changed

to its immediate adjacent values. The neighborhood  $N(p_c)$  is defined by the one-change movement operator which consists of changing the current solution  $p_c$  of a single component either from  $x_{r,i}$  to  $x_{r,i} + 1$  or from  $x_{r,i}$  to  $x_{r,i} - 1$ , where  $r \in R, i \in I_r \setminus \{|I_r|\}$ . Meanwhile, we should guarantee that the sum of the sailing time on each leg plus the duration time on each port is the multiple of seven by adjusting the time of the last component  $x_{r,|I_r|}$ , where  $r \in R$ . Given an incumbent solution  $p_c$ , the one-change movement operator is composed of all possible solutions that can be obtained by applying the one-change movement to  $p_c$ .

#### 6.3.4. Sorted candidate solutions

The candidate solutions  $(SCS_1, SCS_2, \dots, SCS_l, \dots, SCS_{C_{max}})$  are generated after the movement is achieved, where  $C_{max}$  is the number of candidate solutions, and the fitness values of the candidate solutions  $(SCF_1, SCF_2, \dots, SCF_l, \dots, SCF_{C_{max}})$  are sorted in non-increasing order by using the bubble sorting method. Bubble sorting is a simple sort algorithm. It compares two adjacent elements  $SCF_l$  and  $SCF_{l+1}$ . If  $SCF_l$  is less than  $SCF_{l+1}$ , which means their order is opposite, the two adjacent element positions are exchanged and their corresponding candidate solution positions are also updated. If  $SCF_l$  is greater than or equal to  $SCF_{l+1}$ , no transformation operation is taken.

---

**Algorithm 2** Tabu search algorithm for the fleet deployment and demand fulfillment for container shipping liners

---

**Input:** parameters  $T_{ri}^{min}, T_{ri}^{max}, T_{max}, C_{max}, L_{max}, D_{max}, GBF$  //  $P_{ri}^{min}, P_{ri}^{max}$  are the minimum and maximum values of the initial solution with respect to  $r, i$ ;  $T_{max}$  is the given number of iterations for  $t$ ;  $C_{max}$  is the number of candidate solutions;  $L_{max}$  is the tabu list size;  $D_{max}$  is the given number of iterations for  $d$ ;  $GBF$  is the best fitness of all solutions

**Output:** the objective value

```

1: initialization: initial solution  $p = p_0$  //  $p_0$  is the best solution among the  $t$  random solutions
2: neighborhood structure  $N(p)$ 
3: tabu list  $L = \emptyset$ 
4:  $GBF \leftarrow f(p)$ 
5:  $f(p^*) \leftarrow GBF$  //  $p^*$  is the current optimal solution
6:  $p_c \leftarrow p_0$  //  $p_c$  is the incumbent solution
7:  $d \leftarrow 0$  //  $d$  counts the consecutive number of iterations in which  $p^*$  is not improved
8:  $t \leftarrow 0$  //  $t$  counts the consecutive number of iterations where  $p^*$  is not updated
9: while  $t < T_{max}$  do
10: find a best solution  $\hat{p} \in \operatorname{argmax}_{N(p_c)} [f(p_c)]$  //  $\hat{p}$  keeps the best solution found

```

```

551 11:    record the movement in the tabu list
552 12:    if  $\hat{p} \notin L$  then
553 13:        move to the best neighbor  $p_c \leftarrow \hat{p}$ 
554 14:        update tabu list
555 15:    else
556 16:        if  $f(\hat{p}) > f(p^*)$  then
557 17:            move to the best neighbor  $p_c \leftarrow \hat{p}$ 
558 18:             $GBF \leftarrow f(\hat{p}), f(p^*) \leftarrow GBF$ 
559 19:             $p^* \leftarrow \hat{p}, d \leftarrow 0, t \leftarrow 0$ 
560 20:            clean tabu list
561 21:        else if  $f(\hat{p}) \leq f(p^*)$  then
562 22:             $d \leftarrow d + 1$ 
563 23:             $t \leftarrow t + 1$ 
564 24:            if  $d = D_{max}$  then
565 25:                clean tabu list
566 26:                 $sum \leftarrow 0$ 
567 27:                for  $r \in R$ 
568 28:                    for  $i \in I_r \setminus \{|I_r|\}$ 
569 29:                        generate a solution  $sol_{ri}$ , whose value is allocated from  $T_{ri}^{min}$  to  $T_{ri}^{max}$ 
570 30:                         $sum \leftarrow sum + sol_{ri}$ 
571 31:                    end for
572 32:                        adjust  $sol_{r,|I_r|}$  to guarantee  $sum$  is the multiple of seven days
573 33:                end for
574 34:                save the incumbent solution  $p_c \leftarrow (sol_{ri}, r \in R, i \in I_r)$ 
575 35:                 $d \leftarrow 0$ 
576 36:            end if
577 37:        end if
578 38:    end if
579 39: end while
580 40: return the objective value

```

---

### 581 6.3.5. Intensification and diversification strategies

582 The use of memory structures within a tabu search meta-heuristic has been proven  
583 to create a flexible search behavior. A key element of the proposed framework is to  
584 achieve a balance between search intensification and diversification. The intensifica-  
585 tion strategy encourages move combinations and solution features that have appeared  
586 to be effective during the search. In contrast, diversification is used to broaden the  
587 exploration of the solution space. In our algorithm, the diversification strategy clean-  
588 s the tabu list and then randomly generates a new solution. In lines 20 and 25-34  
589 of Algorithm 2, we provide a description of our intensification and diversification

590 strategies.

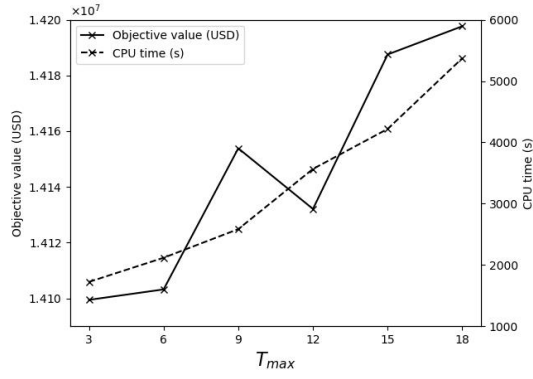
### 591 6.3.6. Sensitivity analysis of the parameters

592 To study the effectiveness of the proposed algorithm, we performed sensitivity  
593 analyses to determine the optimal combination of heuristic parameters. The chosen  
594 four parameters are the consecutive number of iterations where the current optimal  
595 solution is not updated ( $T_{max}$ ), the number of candidate solutions ( $C_{max}$ ), the tabu  
596 list size ( $L_{max}$ ) and the consecutive number of iterations where the current optimal  
597 solution is not improved ( $D_{max}$ ). These parameters are key parameters which may  
598 significantly affect the performance of the tabu search algorithm.

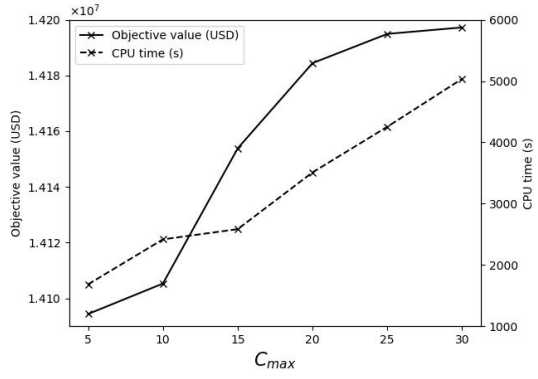
599 To show how the objective value and the computation time are influenced by  
600 parameters  $T_{max}$ ,  $C_{max}$ ,  $L_{max}$  and  $D_{max}$ , we designed four test schemes. The outputs  
601 consist of the computation time and the objective value. When we conduct sensi-  
602 tivity analysis for one parameter, the values of the other three parameters are fixed.  
603 Figure 2-(a) illustrates the interrelation between the value of parameter  $T_{max}$  and  
604 the objective value as well as the computation time, with the value of  $T_{max}$  varying  
605 in  $\{3, 6, \dots, 18\}$ . The same method is applied to parameters  $C_{max}$ ,  $L_{max}$  and  $D_{max}$ ,  
606 varying in  $\{5, 10, \dots, 30\}$ ,  $\{10, 20, \dots, 60\}$ , and  $\{3, 4, \dots, 8\}$ , respectively.

607 The performance of tabu search algorithm is evaluated based on both the objective  
608 value and the computation time. The results in Figure 2 show that with increases in  
609 the values of parameters  $T_{max}$ ,  $C_{max}$ ,  $L_{max}$  and  $D_{max}$ , the computation times of the  
610 tabu search algorithm rise considerably, which indicates that the computation times  
611 are sensitive to the setting of parameters  $T_{max}$ ,  $C_{max}$ ,  $L_{max}$  and  $D_{max}$ . Interestingly,  
612 the objective values of tabu search algorithm grow considerably with the values of  
613 parameters  $T_{max}$  and  $C_{max}$ , but they fluctuate moderately as a function of  $L_{max}$  and  
614  $D_{max}$ , which illustrates that the objective values of tabu search algorithm are sensitive  
615 to the setting of parameters  $T_{max}$ ,  $C_{max}$ , but not to  $L_{max}$  and  $D_{max}$ .

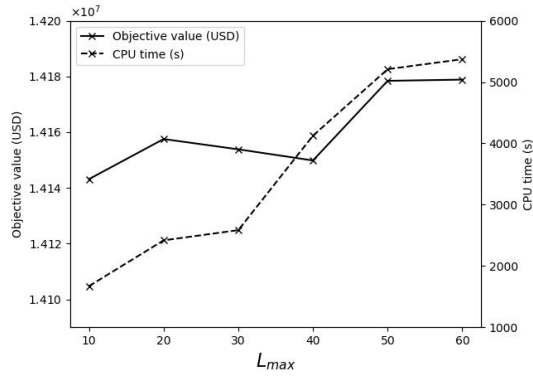
616 We then evaluated the performance of the tabu search algorithm over 36 instances  
617 with fixed  $L_{max} = 20$ ,  $D_{max} = 4$  and different values of  $T_{max}$  and  $C_{max}$ . For each test  
618 instance, several combinations of the two parameters  $T_{max}$  and  $C_{max}$  were used. The  
619 objective values (left column) and the computational time (right column) of each test



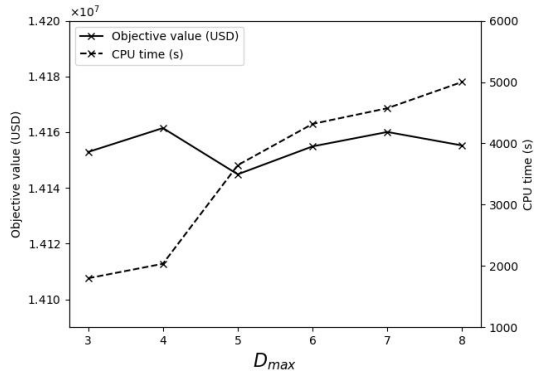
(a)



(b)



(c)



(d)

Figure 2: Sensitivity analysis of the parameters

Table 1: Influence of the parameters  $T_{max}$  and  $C_{max}$  on the performance of the tabu search heuristic

obj and time $T_{max}$	$C_{max}$	5	10	15	20	25	30
4		14,102,397 1,568	14,150,993 1,733	14,159,445 2,518	14,160,098 3,022	14,178,442 3,426	14,192,053 3,677
6		14,140,967 1,821	14,137,802 2,200	14,104,409 2,212	14,135,676 3,454	14,170,465 3,672	14,200,567 4,348
8		14,130,444 1,577	14,143,956 2,417	14,143,855 2,584	14,192,930 3,510	14,137,544 4,255	14,179,581 5,037
10		14,184,778 2,055	14,167,990 2,343	<b>14,210,744</b> 2,555	14,193,398 4,022	14,202,056 4,709	14,188,033 5,633
12		14,150,376 3,734	14,180,336 3,723	14,192,098 3,554	14,197,544 4,392	14,179,002 4,430	14,207,834 5,722
14		14,192,804 4,271	14,192,003 3,356	14,199,581 4,598	14,190,095 4,765	14,201,566 5,221	14,198,892 5,576

**Notes:** (1) The objective values and the CPU time are recorded in the left column and right column, respectively. (2) The objective values and the CPU time are denoted by obj and time, respectively. (3) The CPU time is in seconds.

instance are recorded in Table 1. It can be seen that when  $T_{max} \doteq 10$  and  $C_{max} \doteq 15$ , we can obtain the best results. Therefore, the four values  $T_{max}=10$ ,  $C_{max} = 15$ ,  $L_{max} = 20$  and  $D_{max} = 4$  will be used in the next experiments.

## 7. Computational experiments

In order to assess the effectiveness of the proposed decision model and the efficiency of our algorithms, we have carried out several computational experiments on a LENOVO P910 workstation with 28 cores of CPUs, 2.4 GHz processing speed and 256 GB of memory. All of the models and algorithms proposed in this article were implemented in C# programming. The MIP models (the original model and the submodels embedded in algorithms) were solved by CPLEX 12.5.1.

### 7.1. Instance setting

We first detail the setting of the model parameters. The value of  $V_\varepsilon$  relates to the sailing distance and to the number of containers transported between an OD pair. The sailing distance data can be obtained on the Internet websites, and the unit container revenue data can be acquired on some logistics companies' official websites. The average of  $C_r^{Op}$  is set to 180,000 USD (Wang and Meng, 2015; Wang et al., 2015; Alharbi et al., 2015). The average of  $N_r^{Ship}$  which depends on the length of one cycle time is set to 20. This is consistent with the parameter setting used in previous works (Wang and Xu, 2015; Yao et al., 2012). The average of  $k_{ri}$  is set to 0.25, and the average of  $a_{ri}$  is set to 2.6, which are basically the same as in previous works (Wang et al., 2015; Bell et al., 2013; Yao et al., 2012; Wang and Meng, 2015; Meng et al., 2016). The average of  $C^{Hold}$  is set to 20 USD per day per TEU (Zheng et al., 2015; Wen et al., 2017; Wang and Meng, 2015; Bell et al., 2013). The value of  $\alpha$  is set to 1%. The maximum value of sailing speed is set to 22 knots, which is also in line with the setting used in related works (Jiang and Jin, 2017; Wang et al., 2015; Yao et al., 2012; Aydin et al., 2017). The average of  $C_p^{Berth}$  is set to 3000 per berth (Chen et al., 2012) and the average of  $C_p^{Yard}$  is set to 200 USD per TEU (Jiang and Jin, 2017). The value of  $\bar{D}$  is two days, which is consistent with realistic data from the APL company.

The shipping network investigated in the numerical experiments is depicted in Figure 1. The numbers of routes are three and four in the two different scales of



experiments, and the numbers of ports of call are four, four, five and six in route 1, 2, 3, and 4, respectively. The experimental instances are generated on the basis of a specific rule. Taking the small-scale route network for example, the number of routes is three and the numbers of ports of call are four, four, and five in route 1, 2, and 3, respectively. We can then generate four cases in route 1, which differ from each other only with respect to the ports of call. Each of the four cases uses three ports of call among the four ports of call in the original route 1 shown in Figure 1. Analogously, more sets of cases can be generated through different selections of ports of call in other routes.

Thus for the small-scale without all ports of call network with three routes, there are four sets of cases including three sets without all the ports of call, and an integrated case with all of them. Similarly, as for the large-scale route network consisting of four routes (as shown in Figure 1), there are four sets of cases without all ports of call, and an integrated case with all of them.

## 7.2. Investigating the efficiency of the proposed methods

Here we apply the dynamic linearization algorithm to solve the model  $[M2]$ . A large number of numerical experiments on small-scale cases were carried out to validate this algorithm by comparing the values of its solutions with the optimal results obtained by CPLEX.

From the results shown in Table 2, the objective values obtained by the dynamic linearization algorithm are equal to the optimal results, but this algorithm is faster on the small-scale route network. Based on these observations, we can confirm the efficiency of dynamic linearization algorithm. Table 2 also provides an upper bound (UB) obtained by relaxing Constraints (15), and it shows the gap between the UB and the optimal solution value, which is used to evaluate the efficiency of tabu search algorithm in the large-scale route network. To generate a more complex shipping network, we increase the number of routes from the three to four, which yields a large-scale route network. The results of the experiments show that it is difficult to obtain an optimal solution on this network within a reasonable time.

Table 2: Performance of the dynamic linearization (three routes)

Cases		CPLEX		Dynamic linearization				Upper Bound		
Num. of ports in three routes	ID	$Z_C$	$T_C$	$\pi_\varepsilon$	$Z_D$	$T_D$	$GAP_C$	$\frac{T_D}{T_C}$	$Z_{UB}$	$GAP_{UB}$
<b>3-4-5</b> (Cases differ on the ports in <b>route 1</b> )	Case 1	2,550,670	43	90.34%	2,550,670	13	0.00%	0.30	2,557,281	0.26%
	Case 2	2,592,150	59	92.83%	2,592,150	11	0.00%	0.19	2,605,755	0.52%
	Case 3	2,450,207	28	91.90%	2,450,207	12	0.00%	0.43	2,463,843	0.56%
	Case 4	2,729,982	21	94.55%	2,729,982	8	0.00%	0.38	2,743,593	0.50%
<b>4-3-5</b> (Cases differ on the ports in <b>route 2</b> )	Case 1	2,766,213	48	95.28%	2,766,213	12	0.00%	0.25	2,779,856	0.49%
	Case 2	2,959,825	73	94.46%	2,959,825	17	0.00%	0.23	2,969,885	0.34%
	Case 3	2,307,711	58	93.17%	2,307,711	10	0.00%	0.17	2,308,947	0.05%
	Case 4	2,648,636	27	93.35%	2,648,636	9	0.00%	0.33	2,652,364	0.14%
<b>4-4-4</b> (Cases differ on the ports in <b>route 3</b> )	Case 1	2,354,829	30	91.96%	2,354,829	10	0.00%	0.33	2,368,568	0.58%
	Case 2	2,571,288	56	92.03%	2,571,288	12	0.00%	0.21	2,584,892	0.53%
	Case 3	2,667,825	28	93.30%	2,667,825	9	0.00%	0.32	2,671,536	0.14%
	Case 4	2,570,305	57	92.27%	2,570,305	12	0.00%	0.21	2,576,964	0.26%
	Case 5	2,664,537	35	94.82%	2,664,537	11	0.00%	0.31	2,668,272	0.14%
<b>4-4-5</b>	Case 1	3,905,795	75	93.11%	3,905,795	15	0.00%	0.20	3,921,398	0.40%
<i>Average</i>				<b>93.10%</b>			<b>0.00%</b>	<b>0.28</b>		<b>0.35%</b>

**Notes:** (1) The optimal objective values and the CPU time are denoted by  $Z_C$  and  $T_C$ , respectively. (2) The objective values and the CPU time of the dynamic linearization algorithm are denoted by  $Z_D$  and  $T_D$ , respectively. (3)  $GAP_C = (Z_D - Z_C)/Z_C$ ,  $GAP_{UB} = (Z_{UB} - Z_C)/Z_C$ .

Table 3: Comparing dynamic linearization with tabu search (four routes)

Cases		Dynamic linearization			Tabu search		Comparison	
Num. of ports in four routes	ID	$Z_D$	$Time_D$	$\pi_\epsilon$	$Z_T$	$Time_T$	$GAP_{TD}$	$\frac{Time_T}{Time_D}$
3-4-5-6 (Cases differ on the ports in <b>route 1</b> )	Case 1	11,038,551	3,180	92.02%	11,008,751	1,489	0.27%	0.47
	Case 2	11,874,243	2,692	91.44%	11,809,739	1,803	0.54%	0.67
	Case 3	11,728,095	3,034	93.56%	11,708,295	1,865	0.17%	0.61
	Case 4	12,133,870	1,872	94.47%	12,114,073	1,174	0.16%	0.63
4-3-5-6 (Cases differ on the ports in <b>route 2</b> )	Case 1	12,170,101	2,845	93.23%	12,124,114	1,687	0.38%	0.59
	Case 2	12,470,813	2,923	94.34%	12,421,430	2,075	0.40%	0.71
	Case 3	12,117,924	2,900	93.54%	12,098,124	2,126	0.16%	0.73
	Case 4	12,039,317	2,879	90.86%	11,999,518	2,020	0.33%	0.70
4-4-4-6 (Cases differ on the ports in <b>route 3</b> )	Case 1	11,893,770	3,045	92.88%	11,867,079	1,988	0.22%	0.65
	Case 2	12,178,122	3,300	94.32%	12,145,369	1,723	0.27%	0.52
	Case 3	12,058,507	2,954	95.63%	12,018,702	1,636	0.33%	0.55
	Case 4	11,051,646	2,326	92.32%	11,018,431	1,702	0.30%	0.73
	Case 5	12,069,020	3,875	93.64%	12,029,216	1,734	0.33%	0.45
4-4-5-5 (Cases differ on the ports in <b>route 4</b> )	Case 1	10,947,598	3,004	90.32%	10,927,768	1,556	0.18%	0.52
	Case 2	11,120,029	3,154	91.75%	11,080,223	2,164	0.36%	0.69
	Case 3	11,998,000	2,934	94.92%	11,958,243	2,171	0.33%	0.74
	Case 4	11,594,584	3,357	93.33%	11,554,784	1,947	0.34%	0.58
	Case 5	11,424,056	3,011	92.44%	11,363,650	1,436	0.53%	0.48
	Case 6	11,996,883	2,173	92.46%	11,957,174	1,309	0.33%	0.60
4-4-5-6	Case 1	14,284,795	4,503	90.02%	14,210,744	2,555	0.52%	0.57
<i>Average</i>				<b>92.87%</b>			<b>0.32%</b>	<b>0.61</b>

*Notes:* (1)  $Time_D$  and  $Time_T$  denote the CPU time of the dynamic linearization algorithm and tabu search algorithm, respectively. (2)  $GAP_{TD} = (Z_D - Z_T)/Z_D$ . (3) The CPU time is in seconds.

Therefore, we suggest applying tabu search algorithm to solve the model, and we compare its objective value with that obtained by the dynamic linearization algorithm. The results in the rightmost two columns of Table 3 demonstrate that the average gap between dynamic linearization and tabu search algorithm is about 0.32%, but the average ratio of the CPU time of tabu search algorithm to that of the dynamic linearization algorithm is only 0.61, which indicates that tabu search may not only obtain near-optimal objective function values, but can also solve the model in a much faster way. These results confirm the effectiveness of the dynamic linearization algorithm and of the tabu search algorithm. They demonstrate that tabu search is an effective method for solving the proposed model.

## 8. Conclusions

We have proposed an integrated optimization model for the fleet deployment and demand fulfillment problem, with the consideration of overload risk of containers, vessel size and port resources (e.g., berths, yard space). The objective was to jointly optimize the number of ships in each route, the ship speed on each leg, the visiting time of ships at each port of call, and the fulfillment scale of each OD pair's demand. Since the proposed model is a chance-constrained non-linear MIP model, we have suggested some novel techniques to linearize it into a tractable MISOCP model for some commercial solvers such as the CPLEX. Two efficient algorithms were then suggested to solve the model under different scales of route networks. The proposed model as well as the algorithms can help shipping liners plan the deployment and scheduling of ships along each route. Numerical experiments based on real-word data were conducted to validate the effectiveness of our decision model and the efficiency of the proposed solution methods. With respect to the large body of research on liner ship fleet deployment, we have made three main new contributions:

(1) Few of the previous fleet deployment related studies have considered the demand fulfillment decisions. However, both the fleet deployment and the demand fulfillment decisions are strategic in nature and are intertwined. This study proposed an integrated decision model for optimizing the ship fleet deployment, the scheduling of ship visits at each port of call, and the demand fulfillment scale for each OD pair. The objective was to maximize the total benefit of shipping liners by considering various

711 types of operation costs for running shipping networks.

712 (2) The overload risk of transported containers has seldom been considered in  
713 the FDP related literature, but this issue should not be ignored given the stochastic  
714 weights of containers. Our study takes stochasticity into account by embedding chance  
715 constraints in the decision model so as to control the overload risk under a certain  
716 threshold probability. Some tactics were also suggested to handle the model's non-  
717 linearity as well the complexity yielded by the chance constraints.

718 (3) Several realistic factors ignored in previous studies were considered in our  
719 decision model, but solving them proved to be difficult. We have developed two  
720 algorithms to solve the proposed non-linear chance-constrained MIP on large-scale  
721 instances. Experiments conducted on real-world data demonstrate that our method-  
722 ology yields solutions with an optimality gap less than about 0.5%, and can solve  
723 realistic instances with 19 ports and four routes within about one hour.

## 724 **Acknowledgment**

725 This work was supported by the National Natural Science Foundation of China [grant  
726 numbers 71831008, 71671107, 71422007, 71701178] and the Canadian Natural Sci-  
727 ences and Engineering Research Council [grant number 2015-06189]. Thanks are due  
728 to the reviewers for their valuable comments.

## 729 **References**

- 730 Alharbi, A., Wang, S., Davy, P., 2015. Schedule design for sustainable container  
731 supply chain networks with port time windows. *Advanced Engineering Informatics*  
732 29 (3), 322–331.
- 733 Álvarez, J. F., 2009. Joint routing and deployment of a fleet of container vessels.  
734 *Maritime Economics & Logistics* 11 (2), 186–208.
- 735 Andersson, H., Fagerholt, K., Hobbesland, K., 2015. Integrated maritime fleet de-  
736 ployment and speed optimization: Case study from roro shipping. *Computers &*  
737 *Operations Research* 55, 233–240.
- 738 Aydin, N., Lee, H., Mansouri, S. A., 2017. Speed optimization and bunkering in liner

739 shipping in the presence of uncertain service times and time windows at ports.  
740 European Journal of Operational Research 259 (1), 143–154.

741 Bell, M. G., Liu, X., Angeloudis, P., Fonzone, A., Hosseinloo, S. H., 2011. A  
742 frequency-based maritime container assignment model. Transportation Research  
743 Part B: Methodological 45 (8), 1152–1161.

744 Bell, M. G., Liu, X., Rioult, J., Angeloudis, P., 2013. A cost-based maritime container  
745 assignment model. Transportation Research Part B: Methodological 58, 58–70.

746 Chen, C., Zeng, Q., Zhang, Z., 2012. An integrating scheduling model for mixed  
747 cross-operation in container terminals. Transport 27 (4), 405–413.

748 Cho, S.-C., Perakis, A., 1996. Optimal liner fleet routeing strategies. Maritime Policy  
749 and Management 23 (3), 249–259.

750 Christiansen, M., Fagerholt, K., Nygreen, B., Ronen, D., 2013. Ship routing and  
751 scheduling in the new millennium. European Journal of Operational Research  
752 228 (3), 467–483.

753 Christiansen, M., Fagerholt, K., Ronen, D., 2004. Ship routing and scheduling: Status  
754 and perspectives. Transportation Science 38 (1), 1–18.

755 Cordeau, J.-F., Laporte, G., Legato, P., Moccia, L., 2005. Models and tabu search  
756 heuristics for the berth-allocation problem. Transportation Science 39 (4), 526–538.

757 Fagerholt, K., Johnsen, T. A., Lindstad, H., 2009. Fleet deployment in liner shipping:  
758 a case study. Maritime Policy and Management 36 (5), 397–409.

759 Fransoo, J. C., Lee, C.-Y., 2013. The critical role of ocean container transport in  
760 global supply chain performance. Production and Operations Management 22 (2),  
761 253–268.

762 Gelareh, S., Meng, Q., 2010. A novel modeling approach for the fleet deployment  
763 problem within a short-term planning horizon. Transportation Research Part E:  
764 Logistics and Transportation Review 46 (1), 76–89.

- Glover, F., 1986. Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research* 13 (5), 533–549.
- Jaramillo, D., Perakis, A. N., 1991. Fleet deployment optimization for liner shipping part 2. implementation and results. *Maritime Policy and Management* 18 (4), 235–262.
- Jiang, X. J., Jin, J. G., 2017. A branch-and-price method for integrated yard crane deployment and container allocation in transshipment yards. *Transportation Research Part B: Methodological* 98, 62–75.
- Liu, M., Lee, C.-Y., Zhang, Z., Chu, C., 2016. Bi-objective optimization for the container terminal integrated planning. *Transportation Research Part B: Methodological* 93, 720–749.
- Meng, Q., Du, Y., Wang, Y., 2016. Shipping log data based container ship fuel efficiency modeling. *Transportation Research Part B: Methodological* 83, 207–229.
- Meng, Q., Wang, S., 2012. Liner ship fleet deployment with week-dependent container shipment demand. *European Journal of Operational Research* 222 (2), 241–252.
- Meng, Q., Wang, S., Andersson, H., Thun, K., 2014. Containership routing and scheduling in liner shipping: overview and future research directions. *Transportation Science* 48 (2), 265–280.
- Meng, Q., Wang, T., 2010. A chance constrained programming model for short-term liner ship fleet planning problems. *Maritime Policy and Management* 37 (4), 329–346.
- Meng, Q., Wang, T., 2011. A scenario-based dynamic programming model for multi-period liner ship fleet planning. *Transportation Research Part E: Logistics and Transportation Review* 47 (4), 401–413.
- Meng, Q., Wang, T., Wang, S., 2012. Short-term liner ship fleet planning with container transshipment and uncertain container shipment demand. *European Journal of Operational Research* 223 (1), 96–105.

- 792 Monemi, R. N., Gelareh, S., 2017. Network design, fleet deployment and empty repo-  
 793 sitioning in liner shipping. *Transportation Research Part E: Logistics and Trans-*  
 794 *portation Review* 108, 60–79.
- 795 Nikolopoulou, A. I., Repoussis, P. P., Tarantilis, C. D., Zachariadis, E. E., 2017.  
 796 Moving products between location pairs: Cross-docking versus direct-shipping. *Eu-*  
 797 *ropean Journal of Operational Research* 256 (3), 803–819.
- 798 Perakis, A. N., Jaramillo, D., 1991. Fleet deployment optimization for liner shipping  
 799 part 1. background, problem formulation and solution approaches. *Maritime Policy*  
 800 *and Management* 18 (3), 183–200.
- 801 Petering, M. E., Wu, Y., Li, W., Goh, M., de Souza, R., Murty, K. G., 2017. Real-time  
 802 container storage location assignment at a seaport container transshipment termi-  
 803 nal: dispersion levels, yard templates, and sensitivity analyses. *Flexible Services*  
 804 *and Manufacturing Journal* 29 (3–4), 369–402.
- 805 Powell, B., Perkins, A., 1997. Fleet deployment optimization for liner shipping: An  
 806 integer programming model. *Maritime Policy and Management* 24 (2), 183–192.
- 807 Qi, X., Song, D.-P., 2012. Minimizing fuel emissions by optimizing vessel schedules in  
 808 liner shipping with uncertain port times. *Transportation Research Part E: Logistics*  
 809 *and Transportation Review* 48 (4), 863–880.
- 810 Ronen, D., 1993. Ship scheduling: The last decade. *European Journal of Operational*  
 811 *Research* 71 (3), 325–333.
- 812 Tirado, G., Hvattum, L. M., Fagerholt, K., Cordeau, J.-F., 2013. Heuristics for dy-  
 813 namic and stochastic routing in industrial shipping. *Computers & Operations Re-*  
 814 *search* 40 (1), 253–263.
- 815 Vivaldini, K., Rocha, L. F., Martarelli, N. J., Becker, M., Moreira, A. P., 2016.  
 816 Integrated tasks assignment and routing for the estimation of the optimal number  
 817 of agvs. *The International Journal of Advanced Manufacturing Technology* 82 (1–  
 818 4), 719–736.



- 819 Wang, C., Xu, C., 2015. Sailing speed optimization in voyage chartering ship consid-  
820 ering different carbon emissions taxation. *Computers & Industrial Engineering* 89,  
821 108–115.
- 822 Wang, H., Zhang, X., Wang, S., 2016. A joint optimization model for liner contain-  
823 er cargo assignment problem using state-augmented shipping network framework.  
824 *Transportation Research Part C: Emerging Technologies* 68, 425–446.
- 825 Wang, S., Meng, Q., 2012. Liner ship fleet deployment with container transshipment  
826 operations. *Transportation Research Part E: Logistics and Transportation Review*  
827 48 (2), 470–484.
- 828 Wang, S., Meng, Q., 2015. Robust bunker management for liner shipping networks.  
829 *European Journal of Operational Research* 243 (3), 789–797.
- 830 Wang, S., Meng, Q., Liu, Z., 2013. Bunker consumption optimization methods in  
831 shipping: A critical review and extensions. *Transportation Research Part E: Lo-*  
832 *gistics and Transportation Review* 53, 49–62.
- 833 Wang, S., Wang, T., Meng, Q., 2011. A note on liner ship fleet deployment. *Flexible*  
834 *Services and Manufacturing Journal* 23 (4), 422–430.
- 835 Wang, T., Meng, Q., Wang, S., 2012. Robust optimization model for liner ship fleet  
836 planning with container transshipment and uncertain demand. *Transportation Re-*  
837 *search Record: Journal of the Transportation Research Board* (2273), 18–28.
- 838 Wang, X., Fagerholt, K., Wallace, S. W., 2017. Planning for charters: A s-  
839 tochastic maritime fleet composition and deployment problem. *Omega*, DOI  
840 10.1016/j.omega.2017.07.007.
- 841 Wang, Y., Meng, Q., Du, Y., 2015. Liner container seasonal shipping revenue man-  
842 agement. *Transportation Research Part B: Methodological* 82, 141–161.
- 843 Wen, M., Pacino, D., Kontovas, C. A., Psaraftis, H. N., 2017. A multiple ship routing  
844 and speed optimization problem under time, cost and environmental objectives.  
845 *Transportation Research Part D: Transport and Environment* 52, 303–321.

- 846 Xia, J., Li, K. X., Ma, H., Xu, Z., 2015. Joint planning of fleet deployment, speed  
847 optimization, and cargo allocation for liner shipping. *Transportation Science* 49 (4),  
848 922–938.
- 849 Yao, Z., Ng, S. H., Lee, L. H., 2012. A study on bunker fuel management for the  
850 shipping liner services. *Computers & Operations Research* 39 (5), 1160–1172.
- 851 Yi, W., Chi, H.-L., Wang, S., 2018. Mathematical programming models for construc-  
852 tion site layout problems. *Automation in Construction* 85, 241–248.
- 853 Zacharioudakis, P. G., Iordanis, S., Lyridis, D. V., Psaraftis, H. N., 2011. Liner  
854 shipping cycle cost modelling, fleet deployment optimization and what-if analysis.  
855 *Maritime Economics & Logistics* 13 (3), 278–297.
- 856 Zhao, Y., Jia, R., Jin, N., He, Y., 2016. A novel method of fleet deployment based  
857 on route risk evaluation. *Information Sciences* 372, 731–744.
- 858 Zhen, L., 2015. Tactical berth allocation under uncertainty. *European Journal of*  
859 *Operational Research* 247 (3), 928–944.
- 860 Zhen, L., 2016. Modeling of yard congestion and optimization of yard template in  
861 container ports. *Transportation Research Part B: Methodological* 90, 83–104.
- 862 Zheng, J., Gao, Z., Yang, D., Sun, Z., 2015. Network design and capacity exchange for  
863 liner alliances with fixed and variable container demands. *Transportation Science*  
864 49 (4), 886–899.